Editorial

iMedPub Journals
http://www.imedpub.com/

American Journal of Computer Science and Information Technology
ISSN 2349-3917

2018
Vol.6 No.2:25

DOI: 10.21767/2349-3917.100025

Considerable Detection of Black Hole Attack and Analyzing its Performance on AODV Routing Protocol in MANET (Mobile Ad Hoc Network)

Ashok Koujalagi*

Department of Computer Science, Basaveshwar Science College, Bagalkot, Karnataka, India

***Corresponding author:** Ashok Koujalagi, Department of Computer Science, Basaveshwar Science College, Bagalkot, Karnataka, India, E-mail: koujalagi.ashok@gmail.com

Received date: May 30, 2018; **Accepted date:** June 14, 2018; **Published date:** June 22, 2018

Citation: Koujalagi A (2018) Utilization of edge position for digital image watermarking using discriminant analysis. Am J Compt Sci Inform Technol Vol.6 No.2:25

Abstract

A Mobile Ad hoc Network is an aggregation of mobile terminal that form a volatile network with wireless interfaces. Mobile Ad Hoc Network has no any central administration. MANET more vulnerable to attacks than wired network, as there is no central management and no clear defence mechanism. Black Hole Attack is one of the attacks against network integrity in MANET. In this type of attack all data packets are absorbed by Black Hole node. There are lots of techniques to eliminate the black hole attack on AODV protocol in MANET. In this paper a solution named Black Hole Detection System is used for the detection of Black Hole attack on AODV protocol in MANET. The Black Hole Detection System considered the first route reply is the response from malicious node and deleted, then the second one is chosen using the route reply saving mechanism as it come from the destination node. We use NS-2.35 for the simulation and compare the result of AODV and BDS n solution under Black Hole attack. The BDS solution against Black hole node has high packet delivery ratio as compared to the AODV protocol under Black hole attack and it's about 46.7%.The solution minimize the data loss and decrease the average Jitter 5% and increase the throughput.

Keywords: MANET; AODV; Blackhole AODV; BdsAODV

Overview to Manet (Mobile Ad Hoc Network)

Mobile Ad-Hoc Network is an autonomous system where two or more wireless devices or terminals that has the capability to communicate with each other without of any centralized administrator or fixed network infrastructure. Mobile nodes can dynamically form a network to exchange information without the help of any central administration. MANET are self-organized networks. In MANET mobile nodes are accountable for dynamically discovering other nodes to communicate [1]. Here networks functions like data forwarding, routing, and network administration are carried out synergetic by all available nodes.

In Mobile ad hoc network nodes that are in the radio range can communicate directly, but the nodes that are out of the range can communicate through the intermediate nodes. Nodes are free to move randomly while being able to communicate with each other without the help of an existing network infrastructure. Here mobile node operates not only as an anchor but also as a router for transferring data for other mobile nodes in the network. MANETs are suitable for the situations where any wired or wireless infrastructure is damaged or destroyed [2].

Routing Protocols in Manet

Routing protocol in MANET can be classified into three categories:

Proactive routing protocol (table-driven routing protocol)

In this Routing Protocols, in the network each node must keep up-to-date routing tables. When the network topology changes every node in the network propagates the update message to the network to maintain a reliable routing table. The disadvantages of this routing protocol are that the periodically updating the network topology increases bandwidth overhead and many redundant route entries to the specific destination unnecessarily take place in the routing tables. The advantage is that, if any attacker node joined in network cannot easily attack the network for getting data. Destination Sequenced Distance Vector (DSDV) and Optimized Link State Routing (OSLR), Wireless Routing Protocol (WRP), Global State Routing (GSR) are most familiar types of routing protocols of proactive routing protocol [3].

Reactive routing protocol (on-demand routing protocol)

In reactive routing protocol route tables are created when required and are not maintained periodically. The source node propagates the route request packet to its neighbors when it wants to connect to a destination node. The neighbors of the source node receive the broadcasted request packet and forward the packet to their neighbors until source node's destination is found. After receiving the request the destination node sends a route replay packet to the source node through the shortest path. The path is maintained in the route tables of the nodes through shortest path until the route is no longer needed. This routing protocol is easily affected by the malicious node. Ad-hoc On-demand Distance Vector (AODV), Dynamic Source Routing (DSR) is most familiar routing protocols of active routing protocol [3].

Hybrid routing protocol

The hybrid routing protocol combine the proactive and reactive routing protocols. It is discovered using the advantages of proactive and reactive routing protocols. Hybrid routing protocol uses the proactive routing protocol in the case of intra-domain routing and uses the reactive routing protocol in the case of inter-domain routing. Zone Routing Protocol (ZRP), Temporally-Ordered Routing Algorithm (TORA) is most familiar types of routing protocol of hybrid routing protocol [4].

Ad-Hoc On-Demand Distance Vector (AODV) Routing Protocol

In MANET AODV routing protocol is an on-demand routing protocol used to finding a route to the destination. All mobile nodes work cooperatively to finding route to the destination using the control messages of routing protocol. In AODV routing protocol routes are maintained just as long as it is needed. AODV routing protocol use the destination sequence number for each route entry which is a distinguishable feature from other routing protocol. In AODV routing protocol the routing table stores the destination address, next-hop address, destination sequence number and lifetime. In this, when a node wishes to send a packet to some destination, it checks its routing table to determine if it has a pre-established route to the destination. If it has a pre-established route to the destination, it forwards the packet to next node. If it has not a pre-established route to the destination it launches a route discovery process. For establishing a route to the destination the AODV protocol use the Route Requests (RREQs), Route Replay (RREPs), Route Errors (RERRs) control messages. The source node broadcasts an RREQ message when it wants to established a communication with the targeted node. This RREQ message is inseminated from the source and received by intermediate nodes (neighbors of the source node). When RREQ is received by an transitional node, this fast check its routing table to find a fresh route towards the destination that is requested in RREQ. A route reply (RREP) message is sent towards the source node through the pre-established reverse route (established when RREQ pass through intermediate nodes) if such a route is found. If the transitional node cannot able to find a route, it restore of its routing table and sends RREQ to its neighbors. This action is repeated until the destination nodes receive the RREQ of source node [5]. **Figure1** shows the Route Discovery procedure of Ad Hoc On-Demand Distance Vector Routing where S is the source node and D is the Destination node. Here A, C and B, E are the intermediate nodes for traveling the RREQ message.

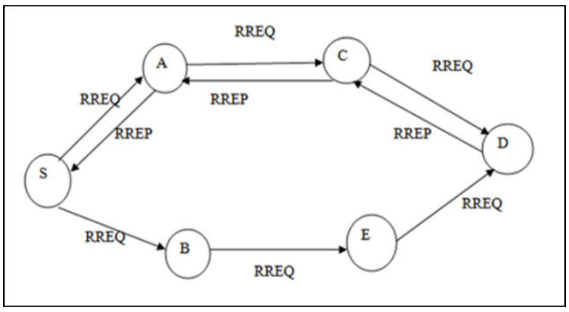


Figure 1: Route Discovery procedure of Ad Hoc On-Demand Distance Vector Routing Protocol.

Every intermediate node in the network increases the hope count one by one when the RREQ (Route Request) packet travels through the network. If a Route Request (RREQ) message is received in a node with the same RREQ ID or Broadcast ID, the receiving intermediate node discard the newly received RREQs with controlling the ID field of the RREQ message. When the RREQ and RREP messages are traveled through the network by the transitional nodes, the transitional nodes update their routing tables and save this route entry for 3 seconds. The ACTIVE_ROUTE_TIMEOUT constant value of AODV protocol is ‘3seconds’ [6].

Black Hole Attack

Mobile Ad Hoc Network using the AODV protocol faces an attack named Blackhole attack where a malicious node or Blackhole node consumes the network traffic and drops all data packets. To explain the Black Hole Attack, an example is shown in the following **Figure2**. In **Figure 2**, we assume that Node B is the malicious node or Black hole node. When Node A broadcasts the RREQ message for Node D to establish a path for data transfer, Node B immediately responds to Node A with a false RREP message showing that it has the highest sequence number of Node D, as if it is coming from Node D. Node A assumes that Node D is behind Node B with 1 hop count and discards the newly received RREP packet come from Node C or E. Node A then starts to send out all data packet to the node B. Node A is trusting that these packets will reach Node D but Node B will drop all data packets. The malicious node or Black hole node takes all the routes coming up to itself. It stops forwarding any packet to any other nodes. The network operation is hampered as the black hole node B consumes the packets easily [7].

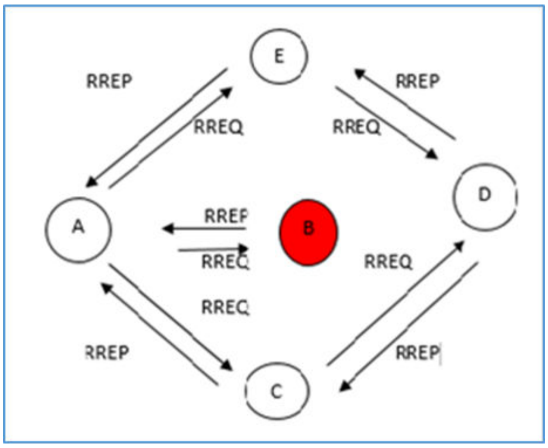
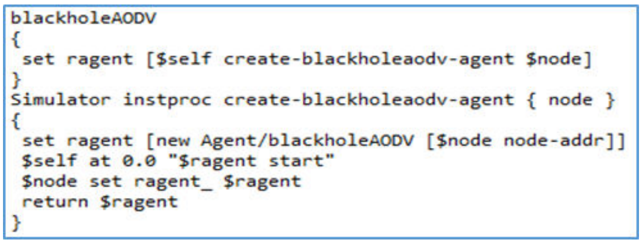


Figure 2: A single Black hole attack in Mobile Ad Hoc Network

Implementing a Routing Protocol in NS to Simulate Black Hole Behaviour

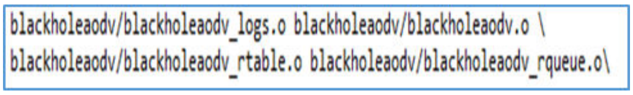
To give a node the characteristics of blackhole node we need to implement a new routing protocol in ns 2.35. Implementation of a New MANET unicast Routing Protocol in NS-2 is described in the reference. [8]. All routing protocols in Network simulator-2.35 are installed in the directory of “ns-2.35”. We first duplicate the AODV protocol in the ns-2.35 directory and change the name of this directory as “blackholeaodv”. In this blackholeaodv directory the name of all files that are labeled as “aodv” are changed to “blackholeaodv” such as blackholeaodv.cc, blackholeaodv.h, blackholeaodv.tcl etc. All classes, functions, variables, and constants names in blackholeaodv directory have changed but struct names that belong to AODV packet.h file have not changed.



```
blackholeAODV
{
 set ragent [$self create-blackholeaodv-agent $node]
}
Simulator instproc create-blackholeaodv-agent { node }
{
 set ragent [new Agent/blackholeAODV [$node node-addr]]
 $self at 0.0 "$ragent start"
 $node set ragent_ $ragent
 return $ragent
}
```

Figure 3: Adding the “blackholeaodv” protocol agent in the “\tcl\lib\ns-lib.tcl” file.

To integrate the new blackholeaodv protocol in NS-2.35 simulator, we have changed two files that are used globally in this simulator. In “\tcl\lib\ ns-lib.tcl” file we first add the lines shown in **Figure-3**, for the agent procedure for blackholeaodv.

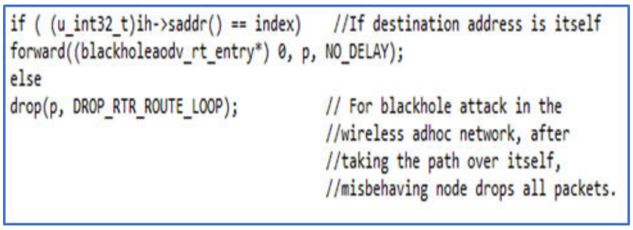


```
blackholeaodv/blackholeaodv_logs.o blackholeaodv/blackholeaodv.o \
blackholeaodv/blackholeaodv_rtable.o blackholeaodv/blackholeaodv_rqueue.o\
```

Figure 4: Addition in the “\makefile” at the ns-2.35 directory

Second file which is in the ns-2.35 directory named “\makefile” where we add the line shown in **Figure 4**.

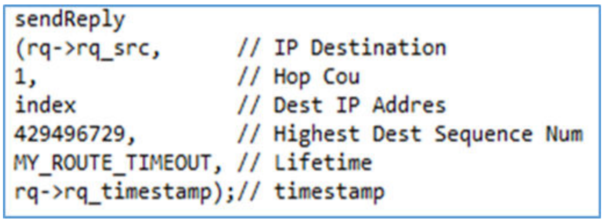
In aodv.cc, the “recv” function process the packet based on the type of the packet. If packet type is AODV route conducting packet such as RREQ, RREP, RERR, it sends the packet to the “recvAODV” function .When the received packet type is data packet type then AODV protocol sends it to the destination address. In the **Figure 5** the first “if” condition provides the node to receive data packets if it is the destination and the “else” condition consume all remaining packets as a Black Hole node.



```
if ( (u_int32_t)ih->saddr() == index)    //If destination address is itself
forward((blackholeaodv_rt_entry*) 0, p, NO_DELAY);
else
drop(p, DROP_RTR_ROUTE_LOOP);          // For blackhole attack in the
                                       //wireless adhoc network, after
                                       //taking the path over itself,
                                       //misbehaving node drops all packets.
```

Figure 5: “If” statement for accepting the packets by destination or dropping packets by malicious node.

To generate the black hole behavior we need to make change in blackholeaodv.cc file by adding the false RREP. The false RREP message show that it has the highest sequence number and the sequence number is set to 4294967295 and hop count is set to 1.The Highest sequence number of AODV protocol is 4294967295, 32 bit unsigned integer value [5]. The lines in **Figure 6** are added to aodv.cc file to generate the characteristics of black hole node. After changing the files then we compiled the “make” in the terminal window (Cygwin window) to create object files.



```
sendReply
(rq->rq_src,      // IP Destination
1,                // Hop Cou
index             // Dest IP Addres
429496729,        // Highest Dest Sequence Num
MY_ROUTE_TIMEOUT, // Lifetime
rq->rq_timestamp);// timestamp
```

Figure 6: The false RREP of blackhole or malicious node.

Solution for the Black Hole Attack on AODV Protocol in Manet

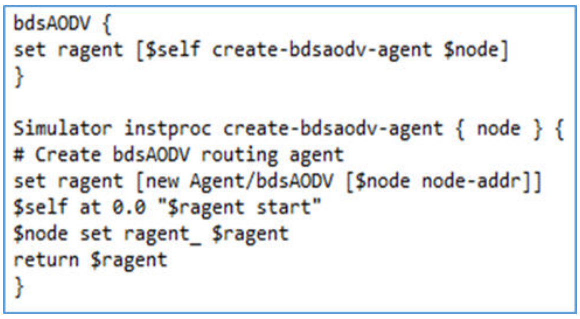
To detect the blackhole attack the “Blackhole Detection System” checks the RREPs that come from multiple paths. As the blackhole node immediately send RREP message to the source without checking its routing table, it is more likely that the first RREP comes from the blackhole node. Then the solution will discard the first RREP packet using the route reply saving mechanism that come from malicious node and choose the second RREP packet.

Algorithm for Black Hole Detection System

- **Step1:** Source node broad cast route Request (RREQ) packet.
- **Step2:** Multiple Route Reply of corresponding Route Request comes to Source node.
- **Step3:** The Route Reply that comes first set as the response from malicious node and removes from the table by using the RREP saving mechanism.
- **Step4:** The second Route Reply is choose by RREP saving mechanism and set it as reply from corresponding destination node. Then the source node delivers the data to the path through which the second RREP came.
- **Step5:** Stop.

Implementing the Black Hole Detection System in NS Against the Black Hole Attack

To implement solution against Blackhole, we duplicated the “AODV” protocol, changing it to “bdsAODV” as we did in “blackholeaodv”. Here for the solution, we had to change the receive RREP function (recv Reply) and create RREP saving mechanism. This RREP saving mechanism counts the second RREP message. At first, we have changed all files name in the cloned “aodv” directory to bdsAODV. To integrate the new bds AODV protocol in NS-2.35 simulator, at First the file “\tcl\lib\ ns-lib.tcl” is modified where protocol agents are coded that is presented in **Figure 7**.

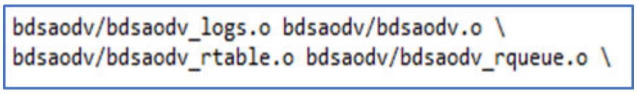


```
bdsAODV {
set ragent [$self create-bdsaodv-agent $node]
}

Simulator instproc create-bdsaodv-agent { node } {
# Create bdsAODV routing agent
set ragent [new Agent/bdsAODV [$node node-addr]]
$self at 0.0 "$ragent start"
$node set ragent_ $ragent
return $ragent
}
```

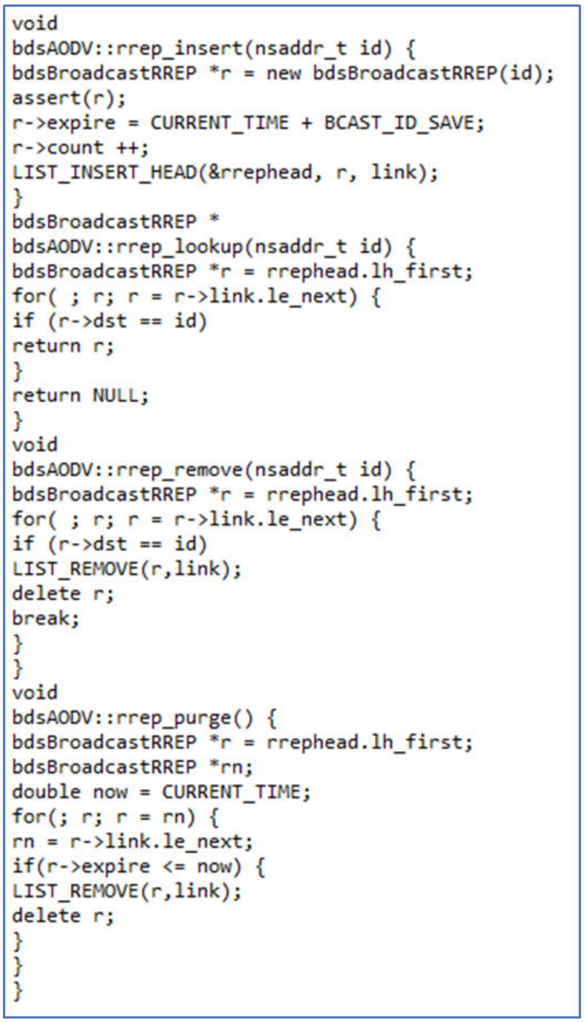
Figure 7: Adding the “proposed” protocol agent in the “\tcl \lib\ ns-lib.tcl” file.

Second file which is in the ns-2.35 directory named “\makefile” where we add the lines that is in **Figure 8**. To detect blackhole attack we create RREP saving mechanism in recv Reply function of bdsAODV.cc file that is presented in **Figure 9**. In the RREP saving mechanism the “rrep_insert” function is used for adding RREP messages, “rrep_lookup” function is used for looking any RREP message up if it is exist,“rrep_remove” function removes any record for RREP message that arrived from defined node and “rrep_purge” function is to delete periodically from the list if it has expired.



```
bdsaodv/bdsaodv_logs.o bdsaodv/bdsaodv.o \
bdsaodv/bdsaodv_rtable.o bdsaodv/bdsaodv_rqueue.o \
```

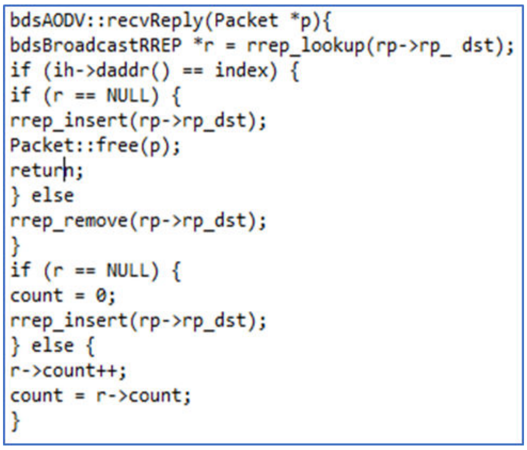
Figure 8: Addition in the “\makefile” at the ns-2.35 directory.



```
void
bdsAODV::rrep_insert(nsaddr_t id) {
bdsBroadcastRREP *r = new bdsBroadcastRREP(id);
assert(r);
r->expire = CURRENT_TIME + BCAST_ID_SAVE;
r->count ++;
LIST_INSERT_HEAD(&rrephead, r, link);
}
bdsBroadcastRREP *
bdsAODV::rrep_lookup(nsaddr_t id) {
bdsBroadcastRREP *r = rrephead.lh_first;
for( ; r; r = r->link.le_next) {
if (r->dst == id)
return r;
}
return NULL;
}
void
bdsAODV::rrep_remove(nsaddr_t id) {
bdsBroadcastRREP *r = rrephead.lh_first;
for( ; r; r = r->link.le_next) {
if (r->dst == id)
LIST_REMOVE(r,link);
delete r;
break;
}
}
void
bdsAODV::rrep_purge() {
bdsBroadcastRREP *r = rrephead.lh_first;
bdsBroadcastRREP *rn;
double now = CURRENT_TIME;
for(; r; r = rn) {
rn = r->link.le_next;
if(r->expire <= now) {
LIST_REMOVE(r,link);
delete r;
}
}
}
```

Figure 9: RREP saving mechanism in the bdsAODV Protocol.

We first check if the RREP message arrived for itself, if it arrived for itself then the function looks up RREP message if it has solution’s receive RREP message function is already arrived. If it did not arrived then it inserts the RREP message for its destination address and returns from the function. If the RREP message is arrived or cached before for the same destination address then the normal RREP function is carried out. If the RREP message is not arrived for itself then the node forwards the message to its appropriate neighbor. The code blocks represented in **Figure10** shown how the bdsAODV carried out.



```
bdsAODV::recvReply(Packet *p){
bdsBroadcastRREP *r = rrep_lookup(rp->rp_ dst);
if (ih->daddr() == index) {
if (r == NULL) {
rrep_insert(rp->rp_dst);
Packet::free(p);
return;
} else
rrep_remove(rp->rp_dst);
}
if (r == NULL) {
count = 0;
rrep_insert(rp->rp_dst);
} else {
r->count++;
count = r->count;
}
```

Figure 10: Receive RREP function of the bdsAODV protocol.

Result Analysis

Here we simulate the same tcl script using AODV protocol without black hole attack, AODV protocol with a single blackhole attack, bdsAODV solution with a single blackhole attack. We simulated our model for 20, 25, 30, 35 and 40 nodes. Then we compared the Performance Metrics such as the PDR (packet Delivery Ratio), End to End delay, through put and Jitter of the three scenarios.

Simulation parameters

Table 1: Simulation parameters

Parameter	Definition
Protocol	AODV, blackholeAODV, bdsAODV
MAC layer	IEEE 802.11
Simulation area	700m*700m
Size of data packet	512 bytes
Traffic sources	CBR/UDP
Number of nodes	20, 25, 30, 35, s40
Number of blackhole node	1
Antenna Type	Antenna/Omni Antenna
Version NS	2.35

Performance evaluation

We have been analyzed the result using four performance metrics. They are the PDR (packet delivery Ratio), End to End Delay, throughput and Jitter (**Table 1**).

Packet delivery ratio (PDR): It is the ratio of the number of data packets received by the destination to the number of data packets generated by the sources node. In the **Figure 11**, PDR values for 20, 25, 30, 35 and 40 nodes for normal AODV, blackholeaodv and bds AODV solution are plotted. Here, it is shown that PDR of AODV is affected by the malicious node and the average packet delivery ratio for this scenario is about 33.42%. Whereas the PDR of bdsAODV against Black Hole node has high packet delivery ratio compared to blackholeaodv and it's about 46.7%.

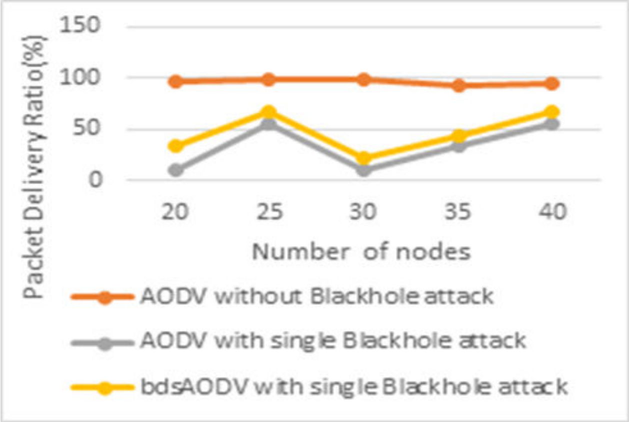


Figure 11: Packet Delivery Ratio (PDR) *vs.* number of nodes.

End to end delay: The time taken by a packet to travel from source to destination is called the End to End delay. In **Figure 12**, the average end to end delay with single Black hole attack is decreased in the bdsAODV solution against Black hole attack.

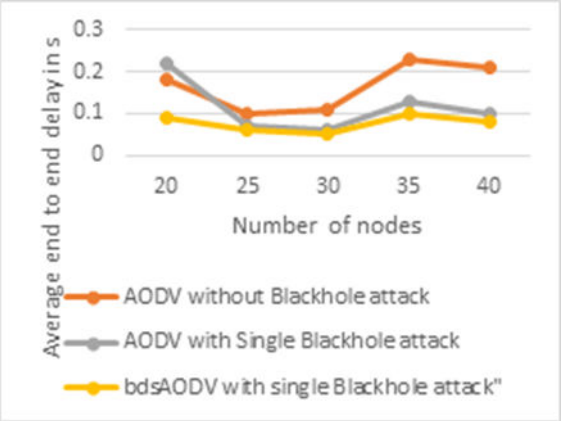


Figure 12: Average End to End Delay *vs.* Number of nodes.

Throughput: Throughput can be defined as the amount of data transferred from sender to receiver in a given amount of time. It is measured in bits per second or packets per second. In **Figure 13**, the bdsAODV solution with single Black Hole attack has average 5kbs-1 throughput which is greater than the average throughput of blackholeaodv that is about 3.6 kb/s^{-1}.

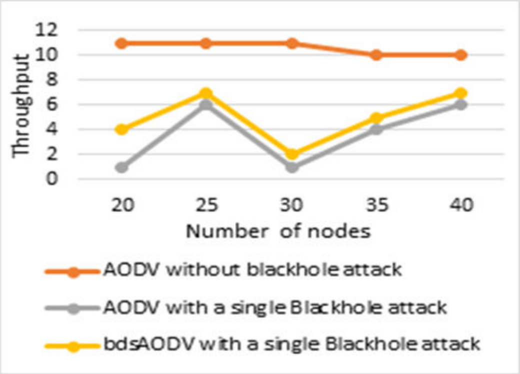


Figure 13: Throughput *vs.* number of nodes.

Jitter: The variation in the delay of received packet is called jitter. Low jitter or Minimum variation in the packet arrival time provides better performance in network. In blackhole scenario, the average jitter is about 12% whereas in bdsAODV solution the average jitter is 7% that means it provide better performance presented in **Figure14**.

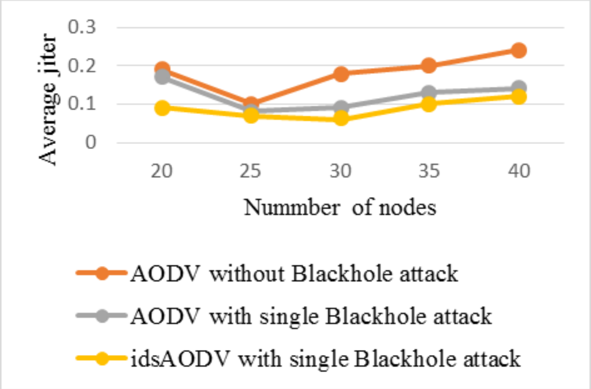


Figure14 : Average jitter *vs.* number of nodes.

Conclusion

When we simulated the Black hole attack, we saw that the data loss is occurred and the packet delivery ratio was decreased. After that, we simulated the bdsAODV solution against black hole attack and saw that the data loss that occurred due to black hole node was decreased. The solution also increased the throughput and decreased the jitter. PDR (Packet Delivery Ratio) of AODV is affected bythe Black Hole node and the average packet delivery ratio for this scenario is about 33.42%. Whereas the PDR of bdsAODV solution against Black hole node has high packet delivery ratio compared to blackholeaodv and it's about 46.7%. The solution decrease the average Jitter 5% compared to the situation of Black hole attack. The advantage of this approach is that for implementing the BDS solution we do not make any modification in packet format hence can work together with AODV protocol. Another advantage is that the solution requires minimum modification in AODV protocol.

References

1. Shrivastava P, Kumar S, Shrivastava M (2014) Study of Mobile Ad hoc Networks. Int J Comp App 86: 0975-8887.
2. Kaur S, Gupta AK (2012) Position Based Routing in Mobile Ad-hoc Networks: An Overview. Int J Comp Sci Technol 3: 792-796.
3. Misra P (2006) Routing Protocols for Ad Hoc Mobile Wireless Networks. http://www.cse.wustl.edu/~jain/cis788-99/adhoc_routing/index.html.
4. Kumar A, Reddy L, Hiremath P (2008) Performance Comparison of Wireless Mobile Ad-Hoc Network Routing Protocols. Int J Comp Sci Net Security 8.
5. Singh G, Singh J (2012) MANET: Issues and Behavior Analysis of Routing Protocols. Int J Adv Res Comp Sci Software Eng 2.
6. Perkins C (2003) (RFC) Request for Comments-3561. Category: Experimental, Network, Working Group.
7. Deng H, Li W, Agrawal DP (2002) Routing Security in Wireless Ad Hoc Networks. University of Cincinnati, IEEE Communication Magazine.
8. Ros FJ, Ruiz PM (2005) Implementing a New Manet Unicast Routing Protocol in NS2. http://masimum.dif.um.es/nsrt-howto/pdf/nsrthowto.pdf.