British Journal of Mathematics & Computer Science
21(2): 1-18, 2017; Article no.BJMCS.31430
ISSN: 2231-0851

SCIENCEDOMAIN *international*
www.sciencedomain.org

A Comparative Analysis of Jellyfish Attacks and Black Hole Attack with Selfish Behavior Attack under AODV Routing Protocol

Bhawna Singla[1*], A. K. Verma[2] and L. R. Raheja[3]

[1]Thapar University, Patiala, India.
[2]CSED, Thapar University, India.
[3]IIT Kharagpur, Kharagpur, India.

Authors' contributions

This work was carried out in collaboration between all authors. Author BS designed the study, performed the statistical analysis, wrote the protocol and wrote the first draft of the manuscript. Authors AKV and LRR managed the analyses of the study. All authors read and approved the final manuscript.

Article Information

DOI: 10.9734/BJMCS/2017/31430
Editor(s):
(1) Qiang Duan, Information Sciences & Technology Department, The Pennsylvania State University, USA.
Reviewers:
(1) Y. Harold Robinson, Anna University, Chennai, India.
(2) Harsh Pratap Singh, Sri Satya Sai Uineristy of Technology and Medical Sciences, Sehore, India.
(3) H. J. Shanthi, AMET University, India.
Complete Peer review History: http://www.sciencedomain.org/review-history/18353

Original Research Article

Received: 5th January 2017
Accepted: 3rd March 2017
Published: 27th March 2017

Abstract

The applications of mobile adhoc network (MANET) are increasing day-by-day due to the flexibility they provide to seamless communication. However MANETS are vulnerable to number of attacks because of properties like non-existing infrastructure, dynamic topology, multihop network etc. Lot of previous works have focused on the impact of various attack on routing protocol. Some attacks like jellyfish attack even follow all the rules and regulations of routing protocol then also they may cause damage to the communication. On the other hand, some attacks like blackhole attack have malicious intentions and causes destruction by dropping the sent packets. There also exist one other category of attack called selfish node attack that do not causes any destruction by modifying the field of the packet rather they do not cooperate in forwarding the packet. In a typical MANET scenario which may be in use for few minutes or even hours, the attacking node will have time to intervene in to the routing process, and able to make some destruction. But, if the network under consideration will be in use for limited short time for a particular military like quick rescue scenario, then how a malicious node will intervene in to the routing

**Corresponding author: E-mail: bhawna_singla@yahoo.com;*

process and make considerable damage to the network within that short duration – this is the research question addressed in this work. In this work we study the impacts of some of the attacks on network under a short term military rescue mission like scenario. We will do a comparative analysis of above discussed attacks under AODV routing protocol. The analysis will be made with respect to different network sizes and under the presence of different number of attackers in the network. The impact on the performance will be measured with suitable metrics to understand the nature of different attacks.

Keywords: AODV; selfish node; black node attack; performance metrics.

1 Introduction

Mobile adhoc network (MANETS) is one of the emerging fields that have seamless applications in the field of emergency situations like rescue operations, commercial applications like virtual classrooms, medical like disease diagnosis etc. MANETs [1] can be describes as collection of nodes in a wireless network where every node is mobile to have dynamic topology. Moreover nodes are even capable of making communications through multiple numbers of nodes where there does not exist any infrastructure to make routing decisions and every node itself acts as a router. In case of indirect communication (i.e. involving number of nodes), path of data packet should follow certain rules and regulation which is determined by routing protocol. Routing protocol [2,3] in case of MANETs is further classified into three categories (i) proactive routing protocol, (ii) reactive routing protocol and (iii) hybrid routing protocol on the basis of route determined on demand or not. Research shows that proactive routing protocol is suitable for that network where topology is less dynamic and reactive routing protocol is suitable for highly dynamic network. As this paper addresses the highly dynamic topology, so we have chosen Adhoc on Demand Distance Vector (AODV) [4] routing protocol for study.

As AODV belongs to the reactive routing protocol, therefore route is determined whenever there is a requirement. Every node maintains the distance vector of chosen metric (like shortest distance or number of hops) for immediate neighbors. Upon requirement, the source node broadcasts the Route Request Control Protocol (RREQ) to its immediate neighbors and waits for the reply. Then there arises two cases:

1. Intermediate neighbor receiving the broadcast message may contain the fresh path to destination or is destination node. If this is the case, then it sends the Route Reply control message (RREP) back to the source node to finalize the path from source to destination.
2. Intermediate node does not contain the fresh path to the destination then it broadcast again the RREQ message to its neighbor node by replacing the IP address field in the header by its address.

As each node has the capability to respond to the control message, therefore AODV is susceptible to number of attacks [4,5,6,7,8,9,10]. When the source node sends a request message for route finding the malicious node responds to it. The malicious node claims to have the shortest route through itself. Once the malicious node is chosen as intermediate node, then it affects the performance of network according to the type of attack. This paper has taken one example from protocol complaint attack called jellyfish attack. Another attack is chosen from malicious attack called blackhole attack and other from passive attack called selfish node attack.

2 Attacks under Investigation

Before actually discussing attacks, a brief description of TCP is presented here. Transmission control protocol (TCP) is a transport layer protocol that is main responsible for i) ordered transmission of packets ii) retransmission of lost packets iii) congestion control. To ensure the delivery of packet when the source node sends a packet, the destination node sends the acknowledgement packet (ACK) back to the source node. Source node maintains the windows of packets for which is awaiting the acknowledgment. A timer is also set

for the maximum time a node should wait for the ACK. If the time exceeds timer, it assumes the packet loss and retransmits the packet. TCP uses 3 way handshake protocols to establish connection.

1. SYN: Whenever the server is ready for connection, the client sends a SYN to the server and sequence number is given a random value A.
2. SYN-ACK: In response to it the server replies by issuing SYN-ACK. The acknowledgement number is given value which is one more then the value of A and sequence number of packet is set to B.
3. ACK: The client sends ACK to the server. The sequence number is set to A+1 and acknowledgement number is set to B+1 thereby ensuring the connection.

   Several extensions to TCP such as TCP Tahoe, TCP Reno, RED, TCP Vegas, New Reno, Sack TCP, Compound TCP have been proposed.

2.1 Jellyfish attack

Jellyfish attack is very much effective where mobility of nodes is more and route lifetime is short. Jellyfish attack by Aad et al. [11] confirms all the protocol rules but degrades the performance of network to near about zero according its three type. Jellyfish attack can be classified into three categories [12,13]: 1) JF reorder attack 2)Periodic Dropping Attack 3)Delay variance attack.

a) JF Reorder Attack: In the jellyfish reorder attack, the packets transmitted sequentially by the source node arrive unordered at the destination. TCP has a vulnerability that the packets once transmitted sequentially may arrive at the destination in unordered sequence due to multipath ordering routing and route changes. Let ACK-N be the cumulative acknowledgement that all the segments from 1,…,N have been received. Then receipt of duplicate ACK-N will show the out of order packets.
b) Periodic Dropping: In this malicious node does not deliver some percentage of packets for maliciously chosen period.TCP throughput can became equal to nearly zero even for the small values of x. Kuzmanovic and Knightly [14] proved that if such losses occur periodically near the retransmission time out (RTO) timescale (in the 1s range as RTO is intended to address severe congestion), then end-to-end throughput is nearly zero.
c) Delay variance attack: TCP assumes the constant round trip time of packet. However due to congestion, the packets have variable round trip time. Such variations may lead to collisions, misestimating of bandwidth and excessively high RTO value. Therefore a malicious manipulation of packet delay reduces the TCP throughput. In Delay Variance attack, the malicious node delays the packet while preserving the order in which packets are transmitted.

2.2 Selfish misbehavior attack

In Selfish node attack [15,16], selfish node aims to save the resources such as CPU time, memory, battery time or bandwidth of network by not forwarding control packets as well as data packets through itself to rest of the nodes. Here if the packet is destined for that selfish node then the packet is accepted otherwise it is not forwarded. However, the node is able to start the communication process. Selfish node attack does not have big impact on the PDR rather they disturb the ETE delays because in presence of selfish node, data packets will be having less option of available routes.

A number of selfishness detection protocols have been invented which can be broadly classified into following categories i) credit based protocol ii) reputation based protocol iii) game theory based protocol.

Credit based Protocols involve the use of virtual or real currency which on the following basis is assigned to node.

1. Data is successfully retransmitted by a intermediate node to other node.
2. Node is properly utilizing the resources available in the network.

Reputation based scheme observes the behavior of node and based on these observations assigns the reputation factor to the node. Based on these reputations, the node is give preference to participate in the future communications.

Game theory based protocol stimulate the game where each node may chose to retransmit the data or not.

2.3 Black hole attack

In black hole problem [17], malicious node claim itself as having the optimal path to the destination node. Once it is chosen as the intermediate node then instead of sending the packet it drops the packet [18,19]. Black hole attack can be caused by RREQ or by RREP. In RREQ black hole attack, attacker pretends to rebroadcast RREQ with non existence node address. As shown in Fig. 3, in RREP black hole attack [20,21,22,23], malicious node can easily send false information regarding the path so as to divert the traffic through itself.

3 Modeling Misbehavior in AODV Routing Protocol

Generally, for implementing different types of attack, it is needed to change several sections of a routing protocol. But in our model, we tried to simplify that. We tried to minimize the "additional lines count", "line change count" while implementing several types of attacks; so that one can easily understand the way in which these kinds of attacks are really working [24].

3.1 Pseudo code of different attacks

The following Pseudo Code in the Fig. 1 Explains the Changes needed in packet forwarding stage of AODV for simulating malicious behavior.

```
forward(Pkt, Delay) {
   if ( ttl=0 ){
      drop(Pkt);
      return;
   }
    if (pkt is addressed to this node)
     recv(pkt);
     return;
    }
if (pkt  is a AODV broadcast) {
    scheduleTransmission(pkt,delay)
    //The Attacks on Aodv pkt is not implemented  here
 } else {
    // here it is a data packet which needs to be forwareded
    If  (AttackMode= none)  {
       scheduleTransmission(pkt,delay)
     else if (AttackMode= JellyfishReorder) {
       //If dest  is me then process the packet normally
            if ( pkt addressed to this node) {
                  scheduleTransmission(pkt,delay)
            } else {
               //here we are imposing reorder
                  //by scheduling the packets at random time
```

```
                    deley= JellyfishReorderLimit * Rand() ;
                        scheduleTransmission(pkt,delay)
            }
        else if (AttackMode= JellyfishPeriodicDropping) {
             //If dst is me then process the packet normally
                if ( pkt addressed to this node) {
                    scheduleTransmission(pkt,delay)
              } else {
                // imposing periodic packet dropping
               //by scheduling the packets at random time
                if (JellyfishAttackProbability > Rand()) {
                   scheduleTransmission(pkt,delay)
                 } else {
                   //Malicious Dropping
                   drop(pkt)
                }
            }
else if (AttackMode= JellyfishDelayVariance) {
           //If dest is me then process the packet normally
              if ( pkt addressed to this node) {
                    scheduleTransmission(pkt,delay)
              } else {
                   //scheduling the packets with high delay
                    deley= JellyfishAttackDelay+ Rand() ;
                        scheduleTransmission(pkt,delay)
            }
 else if (AttackMode= SelfishBehavior) {
            //If dest is me then process the packet normally
              if ( pkt addressed to this node) {
                    scheduleTransmission(pkt,delay)
              } else {
                 //here we are behaving selfishly
                    MaliciousDrop(pkt);
            }
 else if (AttackMode= BlackHole) {
            //If dest is me then process the packet normally
              if ( pkt addressed to this node) {
                    scheduleTransmission(pkt,delay)
              } else {
                 //here we are behaving selfishly
                MaliciousDrop(pkt);
            }
      }
    }
```

Fig. 1. Forward packet function in presence of blackhole attack, selfish node attack and jellyfish attack

In addition to the above function, for blackhole attack, another functions also implemented for generating fake replies to the RREQ messages as shown in the Fig. 2.

```
OnRecieveRReq( Pkt ) {
if (AttackMode= BlackHole) {
 // send fake reply with lesser hop
  sendFakeReply();
  //drop the original request
  drop(Pkt);
else
      if (CurrentNodeIsDestination) {
            //send normal reply
            sendRReply()
      }else {
         //forward pkt normally
            Forward(Pkt)
      }
}
```

Fig. 2. On recieve RREQ function

4 Simulation of Attacks under NS2 Simulator

4.1 Network simulator 2 (NS2)

We used network simulator version NS2.35 under Ubuntu 10.04 operating system for achieving best performance in terms of speed. We have implemented the attacks on the AODV code of NS2.

4.2 The changes made in NS2 AODV code

4.2.1 Changes made in aodv.h

The additional function definitions for simulating attacks and the variables that will be bound with TCL are declared in aodv.h. By using the variables from a TCL simulation code, we can control the behavior of the routing agent.

4.2.2 Changes made in aodv.cc

The actual code of the additional function definitions for simulating attacks were implemented in aodv.cc. And here the new interfaces to the code through the control variables that will be bound with TCL are written here. By setting the variables from a TCL simulation code, we can control the behavior of the routing agent.

4.3 The functions modified for simulating attacks

4.3.1 The function AODV::command(..)

Here the interface to the newly added functionalities are provided. It means we can set some of the variables of C++ code from the TCL simulation script through the interfaces provided in this function.

4.3.2 The function AODV::AODV(..)

In the constructor section of the aodv code, the code needed for binding of new control variables is added.

4.3.3 The function AODV::recvRequest(..)

In this function, the malicious fake route reply code for blackhole attack is implemented. With respect to the value of a control variable “AttackType”, the aodv routing agent will behave normal or malicious.

4.3.4 The function AODV::forward(..)

In this function, the code for different attacks such as Jellyfish Reorder Attack, Jellyfish Periodic Dropping Attack Jellyfish Delay Variance Attack, Selfish Behavior Attack and Black Hole Attack were implemented. With respect to the value of a control variable "AttackType", the aodv routing agent will behave normal or do a particular attack.

After the modifications on aodv.h and aodv.cc, the new version of ns2 is compiled to incorporate the modified version of AODV routing agent. Now the modified version of AODV routing agent can be used in a TCL simulation code. And the functionality of the aodv agent can be controlled by setting up the suitable value in control variable or a using appropriate aodv initialization function that is newly added in AODV::command(..) section.

5 Results and Discussion

5.1 The simulation parameters

In our simulation, we used following common parameters of Table 1 while setting up the network.

Table 1. Simulation parameters

Topographical area	1800 X 500
Mobility	20m/s
Pause time	20s
Total simulation time	100s
Routing protocol	AODV
Mobility model	Random waypoint
Channel model	Wireless channel
Propagation model	Two ray ground
Phy model	Wireless Phy
Mac model	802_11
Antenna model	Omni antenna
Queue	Drop Tail-Pri Queue
Queue length	50

5.2 Traffic parameters

The following parameters of Table 2 are used to setting up the tcp flows with some periodic data.

Table 2. Traffic parameters

Transport agent	TCP
No flows	10
Traffic type	CBR
Packet size	1 Kb
Interval	100 ms
Rate	10 kb

5.3 Variable parameters

The following parameters of Table 3 are used as variables for analyzing the impact of the different attacks on different condition.

Table 3. Table showing total no. of attacking nodes, total nodes and types of attack

Attacking nodes	**5, 10, 15 and 20**
Total nodes	**40, 50, 60**
Simulated attacks	a) Black hole attack b) Selfish behavior attack c) Jellyfish reorder Attack d) Jellyfish periodic dropping attack e) Jellyfish delay variance attack

5.4 Metrics considered for evaluation

In order to evaluate the performance of ad hoc network routing protocols, the following metrics were considered:

5.4.1 Data packets maliciously dropped at routing layer

The count of data packets maliciously dropped at routing layer is the main metric which will help us to understand the malicious behavior of an attack at routing layer.

5.4.2 Total data packets sent

The count of data packets sent at source is a metric which will help us to understand impact of malicious behavior of an attack at routing layer on Application Layer.

5.4.3 Total data packets received

The count of data packets received at destination is a metric which will help us to understand impact of malicious behavior of an attack at routing layer on Application Layer.

5.4.4 Achieved throughput

The throughput is an important metric which will show the impact of attack. Throughput is the number of bytes or bits arriving at the sink over time. It is generally measured in Kbps or Mbps.

5.4.5 Packet delivery fraction (PDF)

Packet delivery fraction is the ratio of the number of packets received at destination to the total number of packets sent from the source.

5.4.6 End-to-end delay

The average time interval between the generation of a packet in a source node and the successfully delivery of the packet at the destination node.

5.4.7 Consumed battery energy

The consumed energy per node is the metric which will show whether the attack caused energy loss. Battery energy consumption is generally measured in Joules.

5.4.8 Dropped packets at source and destination

We considered the packets dropped at source and destination as a metric to measure the impact of attack. Because, it an intermediate attacker causes destruction to a data flow, then it may induce packet loss at source and destination itself.

5.4.8.1 Analytic results with respect to different number of malicious nodes

In the following analysis the total number of nodes in the network is kept as 40 and among them the number of malicious nodes were varied as 10, 15 and 20. And the impact is measured using different metrics.

The following line graph in the Fig. 3 shows the impact of different attacks in terms of total data packets sent at application source. As shown in the line graph, under the presence of Blackhole Attack the application source itself can not able to send much. Selfish Behavior Attack seems to be causing little bit higher impact than all the Jellyfish Attacks. With respect to the increase of no of attackers, the performance decreases.

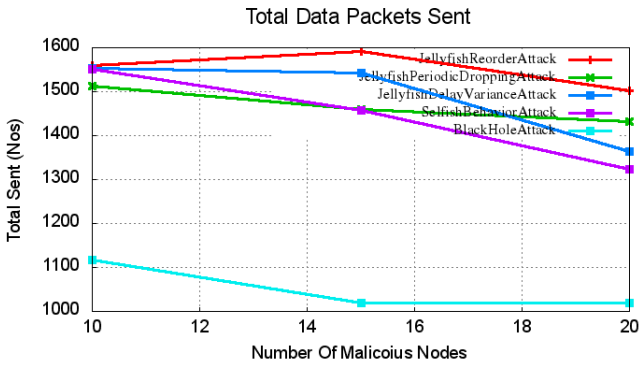

Fig. 3. Attackers vs sent data packets

The following line graph in the Fig. 4 shows the impact of different attacks in terms of total data packets received at application destination. As shown in the line graph, under the presence of Blackhole Attack the application destination can not able to receive much. Selfish Behavior Attack seems to be causing little bit lower higher than all the Jellyfish Attacks. With respect to the increase of no of attackers, the performance decreases. As the blackhole attack mainly focuses on dropping of packet that is why the total sent packet is least in this case. However the performance further decreases with the increase in the number of malicious node.

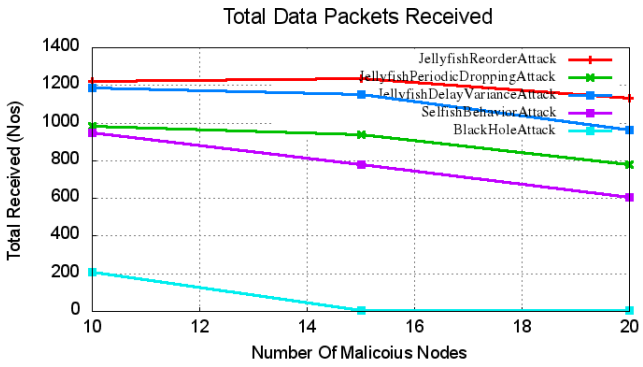

Fig. 4. Attackers vs received data packets

The following line graph in the Fig. 5 shows the impact of different attacks in terms of data packets dropped at source and destination. It signifies the packets dropped at application layer. As shown in the line graph, Blackhole Attack caused much packet dropping at application layer. Selfish Behavior Attack seems to be causing little bit higher impact than all the Jellyfish Attacks. With respect to the increase of no of attackers, the performance slightly decreases. As a significant number of packets have been drooped so the packets received in case of blackhole attack is less than other attacks such as jellyfish and selfish.

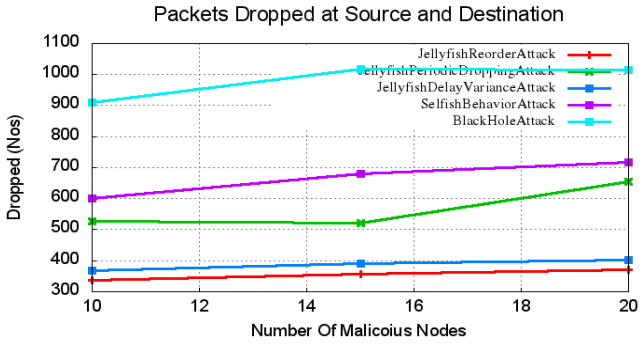

Fig. 5. Attackers vs dropped at application layer

The following line graph in the Fig. 6 shows the impact of different attacks in terms of maliciously dropped data packets at routing layer. As shown in the line graph, except Jellyfish Reorder Attack and Jellyfish Delay Variant attack, all the other attacks maliciously dropping packets at routing layer. Of course, conceptually, Jellyfish Reorder Attack and Jellyfish Delay Variant attack will not drop any packet at routing layer; but only affect the packet transmission/forwarding in different way. The selfish behavior causes little bit of high data packet drop at routing layer. With respect to the increase of no of attackers, the malicious drops at routing layer is getting increase considerably.

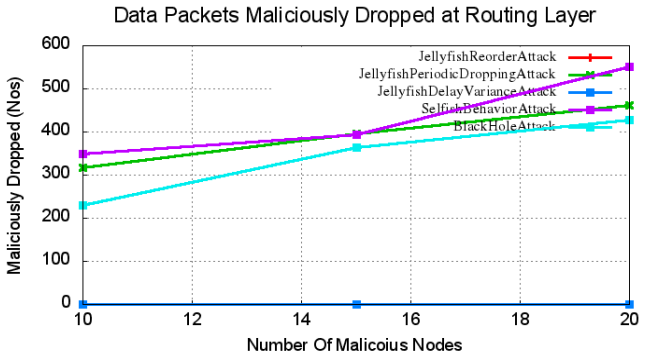

Fig. 6. Attackers vs maliciously dropped at routing layer

The following line graph in the Fig. 7 shows the impact of different attacks in terms of average achieved throughput of tcp flows. As shown in the line graph, Blackhole Attack caused much packet loss so that the throughput was very lower than all other attacks. Next to Blackhole Attack, Selfish Behavior Attack seems to be causing little bit higher impact than all the Jellyfish Attacks. With respect to the increase of no of attackers, the throughput is getting decreased considerably.

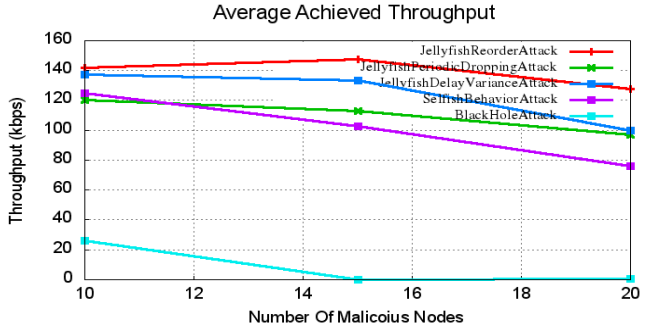

Fig. 7. Attackers vs throughput

The following line graph in the Fig. 8 shows the impact of different attacks in terms of Packet Delivery Fraction (PDF) of tcp flows. As shown in the line graph, Blackhole Attack caused much packet loss so that the PDF was very lower than all other attacks. Next to Blackhole Attack, Selfish Behavior Attack seems to be causing little bit higher impact than all the Jellyfish Attacks. With respect to the increase of no of attackers, the performance decreases.

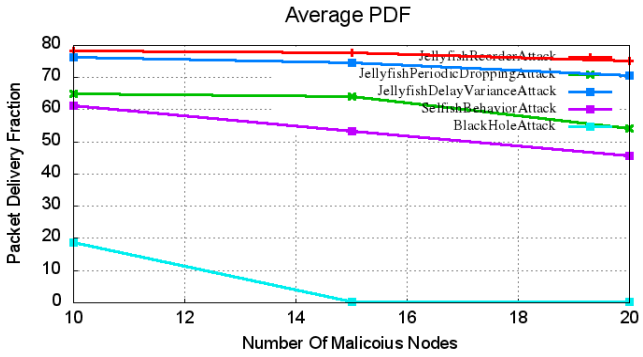

Fig. 8. Attackers vs dropped PDF

In most of the earlier papers, the metrics EED and Energy consumption were not studied in detail or with proper comparison, because, the performance in terms of these two metrics will be somewhat strange to a researcher who always expect worst performance in the presence of an attack. Even in some previous papers, we may found some wrong interpretation for these graphs or even in correctly prepared graphs for these metrics.

The following line graph in the Fig. 9 shows the impact of different attacks in terms of End to End Delay (EED) of tcp flows. As shown in the line graph, Blackhole Attack and Selfish Behavior Attack seems to be providing lower EED than all other Jellyfish Attacks – but certainly it does not mean that Black hole Attack and Selfish Behavior Attack are improving the performance or the network. This of course may be due to number of packets dropped/unforwarded by black hole attack/selfish node attack. With respect to the increase of no of attackers, the performance getting affected with respect to the nature of attack.

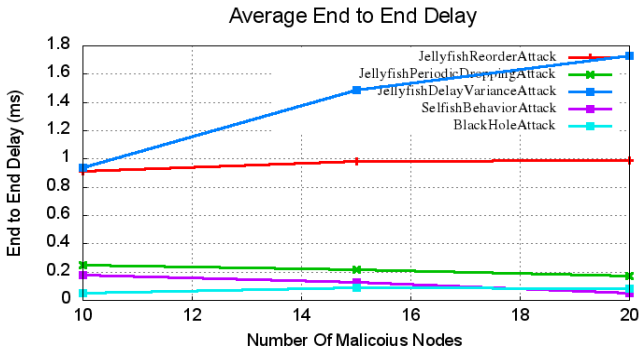

Fig. 9. Attackers vs end to end delay

The low end to end delay under this two attacks are due to a strange fact that these two attacks makes disconnection in tcp flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the message overhead in the network and reduced bandwidth usage otherwise it will be consumed by the forwarded data packets. So, the flows that were unaffected by Blackhole Attack and Selfish Behavior Attack (the connections where there is no neighboring attack nodes) utilizes that extra bandwidth and gains some performance in term of some metrics.

Further, keep in mind that the end to end delay is only calculated based on the time in which a packet is sent and received. So if a packet is not received, in that case end to end delay can not be calculated. So this average EED is only the average EED of successfully delivered packets.

Understanding these strange facts requires a better visualization of the whole network scenario.

The following line graph in the Fig. 10 shows the impact of different attacks in terms of consumed battery energy. As shown in the line graph, in the presence of Blackhole Attack and Selfish Behavior Attack the battery consumption is lesser than all other Jellyfish Attacks – but certainly it does not mean that Black hole Attack and Selfish Behavior Attack are improving the performance in terms of energy consumption. The low energy consumption under this two attacks are due to a strange fact that these two attacks makes disconnection in tcp flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the battery consumption at the other nodes otherwise it will be consumed for forwarding the data packets. So, the nodes that were unaffected by Blackhole Attack and Selfish Behavior Attack (where there is no neighboring attack nodes) preserves some battery power. Understanding this strange fact requires a better visualization of the whole network scenario.

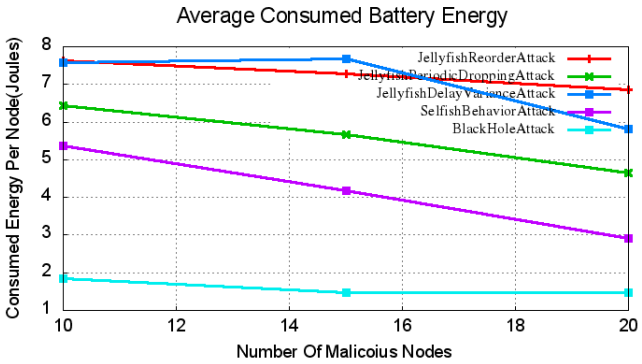

Fig. 10. Attackers vs battery energy

Lot of previous papers saying that the attacks will increase energy consumption. Of course, it also may be true – but not in the same sense. For example if an application will continuously try to send data under attack, then the battery of the sending node and some other nodes between sender and attacker nodes will get reduced rapidly. If the application will vigorously try to do retransmission due to loss, then this will increase the energy consumption. But under tcp, it will handle lossy scenario and just reduce the sending rate to avoid further loss. That is why the average energy consumed in the network seems to be getting reduced. Understanding this strange fact requires a better visualization of the whole network scenario.

5.4.8.2 Analytic results with respect to different network size

Here we see the analytic results of comparision of different attacks with normal AODV (it means performance without any attack). And it is studied with Respect to Different Network Size.

In the following analysis the total number of nodes in the network is varied as 40, 50 and 60 and among them, the number of malicious nodes kept as 20. And the impact is measured using different metrics.

The following line graph in the Fig. 11 shows the impact of different attacks in terms of total data packets sent at application source. As shown in the line graph, under the presence of Blackhole Attack the application source itself can not able to send much. Selfish Behavior Attack seems to be causing almost equal impact like all the Jellyfish Attacks. But even without the presence of any attack AODV performs good and able to send much data packets. With respect to the increase of no of nodes in the network, the performance decreases in most of the cases.

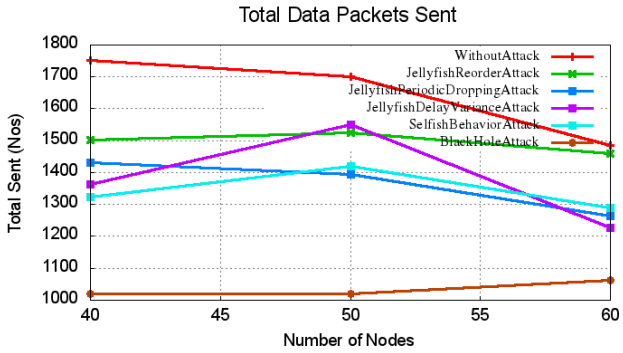

Fig. 11. Network size vs sent packets

The following line graph in the Fig. 12 shows the impact of different attacks in terms of total data packets received at application destination. As shown in the line graph, under the presence of Blackhole Attack the application destination can not able to receive much. Next to Blackhole Attack, Selfish Behavior Attack seems to be causing much impact than all the Jellyfish Attacks. But even without the presence of any attack AODV performs good and able to send much data packets. With respect to the increase of no of nodes in the network, the performance decreases in most of the cases.

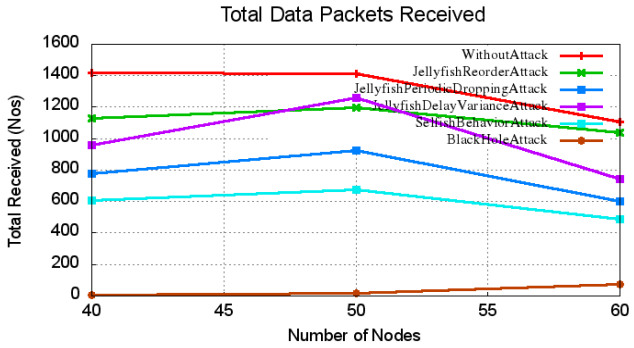

Fig. 12. Network size vs received packets

The following line graph in the Fig. 13 shows the impact of different attacks in terms of data packets dropped at source and destination. It signifies the packets dropped at application layer. As shown in the line graph, Blackhole Attack, Selfish Behaviour Attack and Jellyfish Periodic Packet Dropping Attacks were causing much packet drop at application layer. The other two types of Jellyfish Attacks also causing little packet drop at application layer. But without the presence of any attack AODV performs good and dropping less packets at application layer.

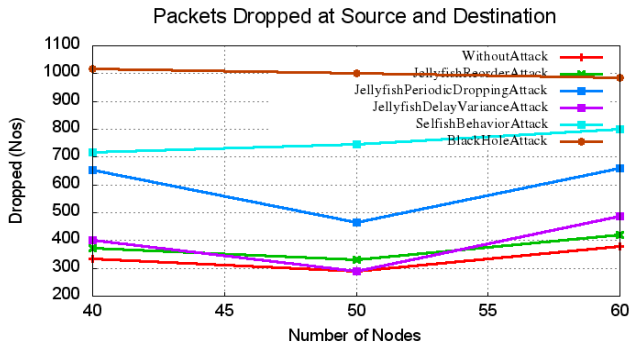

Fig. 13. Network size vs packets dropped at application layer

The following line graph in the Fig. 14 shows the impact of different attacks in terms of maliciously dropped data packets at routing layer. As shown in the line graph, except Jellyfish Reorder Attack and Jellyfish Delay Variant attack, all the other attacks maliciously dropping packets at routing layer. Of course, conceptually, Jellyfish Reorder Attack and Jellyfish [13] Delay Variant attack will not drop any packet at routing layer; but only affect the packet transmission/forwarding in different way. The selfish behavior causes little bit of high data packet drop at routing layer. With respect to the increase of no of nodes in the network, the malicious dropping increasing a little bit.

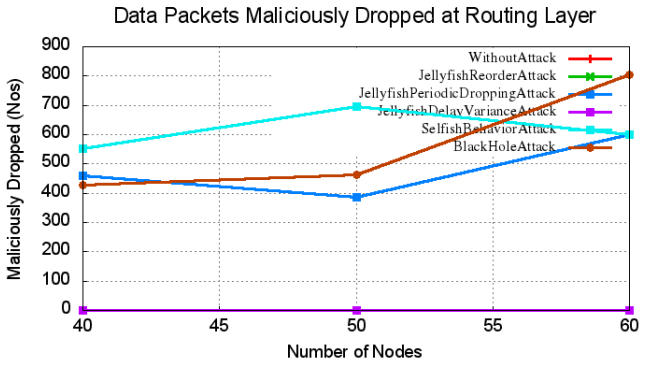

Fig. 14. Network size vs malicious drops at routing layer

The following line graph in the Fig. 15 shows the impact of different attacks in terms of average achieved throughput of tcp flows. As shown in the line graph, Blackhole Attack caused much packet loss so that the throughput was very lower than all other attacks. Next to Blackhole Attack, Selfish Behavior Attack seems to be causing little bit higher impact than all the Jellyfish Attacks. But without the presence of any attack AODV performs good and provided highest throughput. With respect to the increase of no of nodes in the network, the throughput decreases considerably.

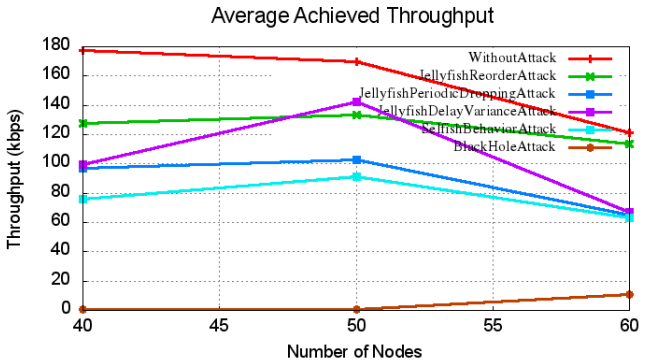

Fig. 15. Network size vs throughput

The following line graph in the Fig. 16 shows the impact of different attacks in terms of Packet Delivery Fraction (PDF) of tcp flows. As shown in the line graph, Blackhole Attack caused much packet loss so that the PDF was very lower than all other attacks. Next to Blackhole Attack, Selfish Behavior Attack seems to be causing little bit higher impact than all the Jellyfish Attacks. But without the presence of any attack AODV performs good and provided highest PDF. With respect to the increase of no of nodes in the network, the performance getting decreased in most of the cases. But in the case of Blackhole attack, with respect to the increase of no of nodes in the network, the performance getting increased because with the high number of nodes, there was chances for developing alternate path that may avoid malicious nodes in it.

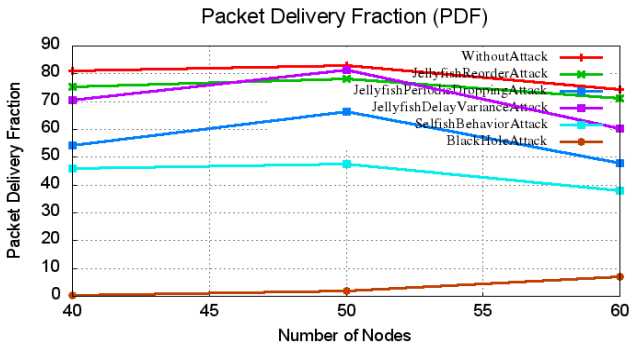

Fig. 16. Network size vs PDF

In most of the earlier papers, the metrics EED and Energy consumption were not studied in detail or with proper comparison, because, the performance in terms of these two metrics will be somewhat strange to a researcher who always expect worst performance in the presence of an attack. Even in some previous papers, we may found some wrong interpretation for these graphs or even in correctly prepared graphs for these metrics.

The following line graph in the Fig. 17 shows the impact of different attacks in terms of End to End Delay (EED) of tcp flows. With respect to the increase of no of nodes in the network, the performance getting decreased .As shown in the line graph, Blackhole Attack and Selfish Behavior Attack seems to be providing lower EED normal AODV(without attack) – but certainly it does not mean that Black hole Attack and Selfish Behavior Attack are improving the performance or the network. The low end to end delay under this two attacks are due to a strange fact that these two attacks makes disconnection in tcp flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the message overhead in the network and reduced bandwidth usage otherwise it will be consumed by the forwarded data packets. So, the flows that were unaffected by Blackhole Attack and Selfish Behavior Attack (the connections where there is no neighboring attack nodes) utilizes that extra bandwidth and gains some performance in term of some metrics.

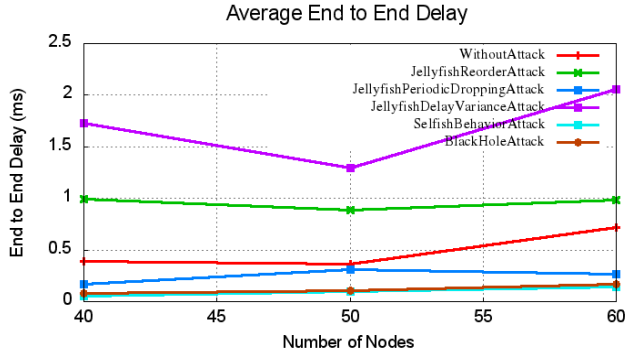

Fig. 17. Network size vs end to end delay

Further, keep in mind that the end to end delay is only calculated based on the time in which a packet is sent and received. So if a packet is not received, in that case end to end delay can not be calculated. So this average EED is only the average EED of successfully delivered packets.

The following line graph in the Fig. 18 shows the impact of different attacks in terms of consumed battery energy. As shown in the line graph, in the presence of all the kinds of Attack the battery consumption is lesser than Normal AODV (without attack) – but certainly it does not mean these Attacks are improving the performance in terms of energy consumption. The low energy consumption under attacks are due to a

strange fact that these attacks makes disconnection in tcp flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the battery consumption at the other nodes otherwise it will be consumed for forwarding the data packets. So, the nodes that were unaffected by Attacks (where there is no neighboring attack nodes) preserves some battery power. Understanding this strange fact requires a better visualization of the whole network scenario. It is simple – without any attack, AODV was able to send much and maximum nodes were able to participate in that communication and utilized their energy for transmission/forwarding of packets – so that the energy is consumed in most of the nodes. But in the presence of attack, the packets are getting dropped intermediately and the battery powers on other nodes that are not at all forwarding the packets gets preserved. With respect to the increase of no of nodes in the network, the performance seems to be getting decreasing.

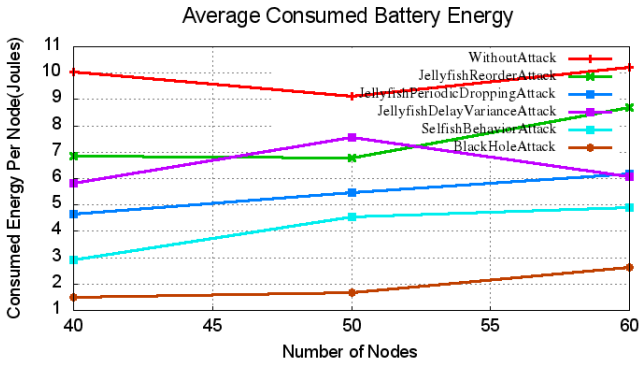

Fig. 18. Network size vs battery energy

Lot of previous papers saying that the attacks will increase energy consumption. Of course, it also may be true – but not in the same sense. For example if an application will continuously try to send data under attack, then the battery of the sending node and some other nodes between sender and attacker nodes will get reduced rapidly. If the application will vigorously try to do retransmission due to loss, then this will increase the energy consumption. But under tcp, it will handle lossy scenario and just reduce the sending rate to avoid further losss. That is why the average energy consumed in the network seems to be getting reduced under attack. Understanding this strange fact requires a better visualization of the whole network scenario.

6 Conclusion

In this work we studied the impacts of some of the popular attacks on a short term military rescue mission like MANET scenario. We did a comparative analysis of three kinds of Jellyfish Attacks and Black Hole Attack with Selfish Behavior Attack under AODV routing protocol and presented our findings. We did that analysis with respect to different network sizes and under the presence of different number of attackers in the network. We did lot of simulation and analysis and arrived significant and interpretable results. We measured the impact of the attacks with suitable metrics and explained the nature of different attacks in the previous chapter. With respect to the increase of malicious nodes in the network, the performance is getting decreased with respect to the most of the metrics that we considered. Further, with respect to the increase in number of nodes in the network, the performance is getting affected with respect to the nature of attack. Without any doubts, all the attacks affects the performance of MANET and the tcp flows are very much affected by all these attacks.

The main scope of this paper it to compare the Selfish Behavior Attack with different Jellyfish Attacks and Blackhole Attack. We successfully did that and the results are more interesting. According to our observations and the arrived results, the Selfish Behavior Attack was as almost worst as Blackhole Attack and even much worst than all types of Jellyfish Attacks with respect to most of the metrics. The future scope of this paper is the by understanding the natures and impacts of attack on routing paper, one can further

devise a attack free solution for the system. The study can also help in devising the security architecture for the routing protocol that is not able to fight against single attack rather all possible attacks.

Competing Interests

Authors have declared that no competing interests exist.

References

[1] Imrich Chlamtac, Marco Conti, Jennifer JN. Liu: Mobile ad hoc networking: Imperatives and challenges. Ad Hoc Networks. 2003;1(1):13-64.

[2] Abolhasan Mehran, Tadeusz Wysocki, Eryk Dutkiewicz. A review of routing protocols for mobile ad hoc networks. Ad Hoc Networks. 2004;2(1):1-22.

[3] Royer Elizabeth M, Chai-Keong Toh. A review of current routing protocols for ad hoc mobile wireless networks. Personal Communications, IEEE. 1999;6(2):46-55.

[4] Perkins Charles, Elizabeth Belding-Royer, Samir Das. Ad Hoc on-demand Distance Vector (AODV) routing. No. RFC 3561; 2003.

[5] Lego Kapang, Dipankar Sutradhar. Comparative study of Ad Hoc routing protocol AODV. DSR and DSDV in Mobile Ad Hoc Network 1; 2011.

[6] Kumar Sathish Alampalayam. Classification and review of security schemes in mobile computing. Wireless Sensor Network. 2010;2(06):419-440.

[7] Desilva Saman, Rajendra V. Boppana. Mitigating malicious control packet floods in ad hoc networks. Wireless Communications and Networking Conference, 2005 IEEE. 2005;4.

[8] Marti Sergio, et al. Mitigating routing misbehavior in mobile ad hoc networks. Proceedings of the 6th Annual International Conference on Mobile Computing and Networking. ACM; 2000.

[9] Wu Bing, et al. A survey of attacks and countermeasures in mobile ad hoc networks. Wireless Network Security. Springer US. 2007;103-135.

[10] Deng Hongmei, Wei Li, Dharma P. Aggrawal. Routing security in wireless ad hoc networks. Communications Magazine, IEEE. 2002;40(10):70-75.

[11] Aad Imad, Jean-Pierre Hubaux, Edward W. Knightly. Denial of service resilience in Ad hoc networks. Proceedings of the 10th Annual International Conference on Mobile Computing and Networking. ACM; 2004.

[12] Mandala Satria, et al. Investigating severity of blackhole attack and its varianace in wireless mobile Ad hoc networks. International Journal of Embedded Systems. 2015;7(3-4):296-305.

[13] Azer Marianne A, Noha Gamal El-Din Saad. Prevention of multiple coordinated Jellyfish attacks in Mobile Ad hoc Networks. International Journal of Computer Applications. 2015;120:20.

[14] Kuzmanovic Knightly. Low-rate TCP-targeted denial of service attacks: The shrew vs. the mice and elephants. Proceeding SIGCOMM '03 Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications. 2003;75-86.

[15] Jo Minho et al. Selfish attacks and detection in cognitive radio ad-hoc networks. IEEE Network. 2013;27(3):46-50.

[16] Wu Bing, et al. A survey of attacks and countermeasures in mobile ad hoc networks. Wireless Network Security. Springer US. 2007;103-135.

[17] Chavan AA, Kurule DS, Dere PU. Performance analysis of AODV and DSDV routing protocol in MANET and modifications in AODV against Black Hole Attack. Procedia Computer Science. 2016;79:835-844.

[18] Laxmi Vijay, et al. JellyFish attack: Analysis, detection and countermeasure in TCP-based MANET. Journal of Information Security and Applications. 2015;22:99-112.

[19] Baadache Abderrahmane, Ali Belmehdi. Struggling against simple and cooperative black hole attacks in multi-hop wireless ad hoc networks. Computer Networks. 2014;73:173-184.

[20] Singh Pramod Kumar, Govind Sharma. An efficient prevention of black hole problem in AODV routing protocol in MANET. 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications. IEEE; 2012.

[21] Virmani Deepali, Pranav Gupta. Adaptive exponential trust-based algorithm in wireless sensor network to detect black hole and gray hole attacks. Emerging Research in Computing, Information, Communication and Applications. Springer Singapore. 2016;65-73.

[22] Salem Abdul-Rahman, Rushdi Hamamreh. Efficient mechanism for mitigating multiple black hole attacks in Manets. Journal of Theoretical and Applied Information Technology. 2016;83(1):156.

[23] Singh Bikramjeet, Dasari Srikanth, Suthikshn Kumar CR. Mitigating effects of Black hole Attack in Mobile Ad-hoc Networks: Military perspective. Engineering and Technology (ICETECH), 2016 IEEE International Conference on. IEEE; 2016.

[24] Ayash Mohannad, Mohammad Mikki, Kangbin Yim. Improved AODV routing protocol to cope with high overhead in high mobility MANETs. Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2012 Sixth International Conference on. IEEE; 2012.

Peer-review history:
The peer review history for this paper can be accessed here (Please copy paste the total link in your browser address bar)
http://sciencedomain.org/review-history/18353