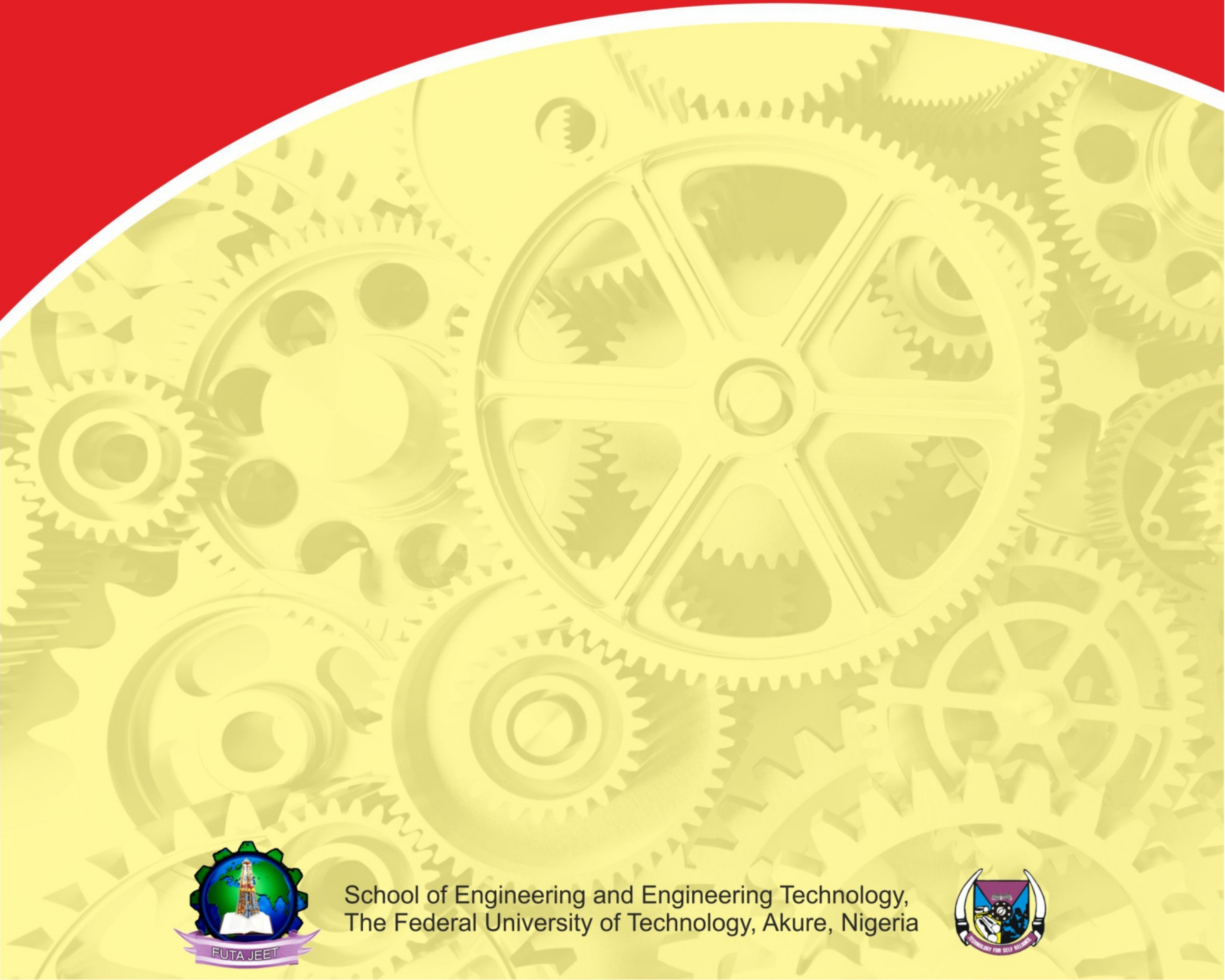
FUTA JEET Vol 10 Issues 1 & 2 December 2016

Journal of Engineering and Engineering Technology

ISSN 1598-0271



School of Engineering and Engineering Technology,
The Federal University of Technology, Akure, Nigeria

Journal of Engineering and Engineering Technology

Performance Evaluation of A Reverse Aodv And Aodv Routing Protocol Under Jellyfish Attacks in Mobile Ad Hoc Network

Azeez M.A., Oluwaseyi O., Iwayemi A., Ajiboye A.S.

Mcal Engineerechaniing Department, The Federal University of Technology, Akure, Ondo State, Nigeria.

Key words:
RAODV,
AODV,
Jellyfish Attack,
Manet,
Protocol

ABSTRACT

Mobile Ad hoc network (MANET) is an example of mobile wireless network characterized by dynamic topology, self-organization, auto-configuration, lack of infrastructure and decentralize administrator that made it more useful in military battlefield, smart building and emergency rescue. The nodes vulnerability in MANET security becomes the center of creative research efforts. Many security schemes were proposed, rather than trying to encompass the entire field of MANET security. In this paper the same security issue was addressed by evaluating the performance of two reactive protocols Ad-hoc On-demand Distance Vector (AODV) protocol and Reverse Ad-hoc On-demand Distance Vector under jellyfish attacks. This research adopts the use of Network simulator 2.35 to test the two reactive protocols vulnerability in the presence of jellyfish attacks. The results of the performance metrics shown that the two protocols average throughputs were affected in the presence of the attacks and increase tremendously as the number of node increases, while packet delivery ratio decrease with increase in number of malicious nodes. The jellyfish attacks have great effect in number of packet drop in the protocols. Less packet drop in the presence of single jellyfish but greatly increase as more threats were introduced and maintain the increase rate as the number of attacks increase. The experimental results for all scenarios shown that the higher the jellyfish attacks on AODV and RAODV the lower the performance of the two protocols.

1. Introduction

Rapid migrations from fixed wireless to mobile wireless network not only give chance to mobile communication but also provides platform for easier exchange of data. MANET as an example of mobile wireless network characterized by dynamic topology, self-organization, auto-configuration, lack of infrastructure and decentralize administrator that made it more useful in military battlefield, smart building and emergency rescue. Each mobile device in a network serves as a node in MANET; every node moves arbitrarily and behaves as router and host. The interconnections between nodes are capable of changing continually based on assume topology and relays other nodes which are far apart from the range to form a communication channel.

(Khetmal, 2013) routing in ad hoc network is to set up minimum pathway between source and destination. This is done with routing protocols. The routing protocols in MANET could be categorized based on the routing topology which are proactive, reactive and hybrid. Proactive protocols also known as table driven i.e. periodically update the routing information to all network nodes.

Update is done regardless of using the route or not, which results to over usage of resources such as energy and bandwidth. AODV as an example of reactive routing protocols. It uses Route Request (RREQ) and Route Reply (RREP) control messages in Route Discovery phase. AODV broadcast RREQ to all the nodes within the range. Each node will reply with RREP to set up route for packet to propagate, but dynamic characteristic of nodes in the range makes the link between the nodes to be transitory. When node sends packet, such packet may lost before it gets to a destination node. This loss of packet affects the performance of route link setup by the source node. The loss of RREP packet creates problems because source node needs to re-establish route discovery process before packet can be sent. This reduces the performance of MANET. Reverse AODV (RAODV) routing protocol, a modified AODV routing protocol, solved this problem. In this protocol,

Correspondence:
E-mail address:azmudray1@gmail.com, iwayemi_ayodeji@yahoo.com

destination node uses the same discovery technique to send a Reverse RREQ(R-RREQ) to find the source node, this increases the network performance and reduces the link failure between the nodes regardless of dynamic topology of the network.

RAODV generates a routing path while sending packet. At this period it creates expire time to sustain its routing table. So, if there is change in network topology, the routing path remains static. If a new shortest path is discovered than previous path during expire time, it does not adjust the routing path because RAODV needs to keep to this path within expire time. The unicast nature of AODV protocol makes it vulnerable to jellyfish attack. Jellyfish is a Denial of Service (DoS) attack, which occurs when malicious node take path in network to accept packet and deliberately drops, delay or alter packet traffic passing through it.

The parameters that were considered for performance metrics are packet delivery ratio, end to end delay, throughput, routing packet sent and routing overhead. (Sanabani et al, 2014) state that RAODV is created to solve the problem of rapid change in topology of MANET. (Biswas et al., 2007) stated that more sophisticated and subtle attacks have been identified in MANET. (Tanwar, et al, 2013) stated that MANET are highly susceptible to routing attacks because of their dynamic topology and lack of any infrastructure. (Rehman, 2010) in black hole attack, a malicious node uses its routing protocol in order to advertise itself for having the shortest path to the destination node or to the packet it wants to intercept. (Sani and Kumar, 2010) stated that the black hole affects the AODV protocol and the effect on packet loss is much lower as compare to effect on delay. (Esmaili et al, 2011) OPNET simulated results depicts that packet delivery ratio in the presence of malicious nodes, reduce notably. (Kaswan and Gupta, 2013) concluded from their simulation result that the black hole attack is more effective in MANETs as compared to the wormhole attack.

Many security schemes have been proposed, rather than trying to encompass the entire field of MANET security, this paper mainly compared the performance evaluation of AODV protocol and RAODV under jellyfish attacks.

2. Methodology

2.1 Simulation Environment

NS-2 Generic Script Structure:

- Create Simulator object
- [Turn on tracing]
- Create topology
- [Setup packet loss, link dynamics]
- Create routing agents
- Create application and/or traffic sources
- Post-processing (nam)
- Start simulation

2.2 Simulation Study

The two protocol simulated in Ns2.35 are AODV and RAODV to set a comparison of how better/worse the protocols perform under jellyfish attacks.

2.2.1 Simulation System Requirement

The simulation was carried out with the following system specification:

- Processor IntelCore i3 – 2330M CPU @ 2.20GHz 2.20Ghz
- Machine: Oracle VM VirtualBox Version 5.0.10
- Base Memory: 1024MB
- System Type: Ubuntu (32bit)
- Installed Operating System: Ubuntu 15.04
- Video memory: 12MB

2.2.2 Simulation Parameters

Figure 1 shows how fixed parameters were declared in the .tcl file. The parameters declared are MacType, Antenna Type, Network Type, Link Layer and Channel Type. The parameters declared is constant throughout the experiment.

2.3 Experimental Setup

The simulation scenario and parameters that were used for performing the detailed analysis of jellyfish attacks on AODV and RAODV routing protocols are stated below:

- Input to Simulator:-
- Scenario File – movement of nodes
  - Traffic pattern file
  - Threat Script Modification
  - Simulation TCL file
- Output File from Simulator
  - Trace File
  - Network Animator file

2.3.1. Simulation Setup

Table 1.2 States the methods for simulation setup

Four techniques were used to carry out the simulation process which are analyzed below

- Mobility Simulation: Creating nodes movement for wireless scenario
- Application of Threats: Increase in numbers of threats where number of nodes are kept constant
- Simulation and observation of each node on the protocols
- Generating and comparison of result for each protocol using the set aside parameters on these metrics (Packet Delivery Ratio, Throughput)

Randomized scenarios with different parameters that affect the movement pattern of nodes were used. The parameters that can be changed are:

- *Maximum Speed:* Every time speed is going to be randomized it is randomized in the internal [0, maximum speed].
- *No of nodes:* Maximum number of nodes used in the simulations procedure is 25
- *Environment Size:* This determined the area of the simulation. 500(x)meter x 500(y)meter was used throughout the simulation
- *Simulation Time:* The simulation was carried out at a constant simulation time of 200seconds.
- *Pause Time:* Pause Time is the time a node stand still before randomizes a new destination and the speed that will be used to reach the destination. Pause time was 10s throughout the simulation.
- *Increase in number of threats:* The nodes that act as malicious (jellyfish) were also increased to check the effect of large numbers of threats in each protocol.

2.5 *Jellyfish Nodes Size Simulation*

Adding a malicious node in ns2 using protocol, the node which is declared as malicious will simply drop the router packet (DROP_RTR_ROUTE_LOOP).

Two files have to be modified.

1. aodv.h, raodv.h
2. aodv.cc, raodv.cc

Once done, ns2 was recompiled with these commands:

```
$] cd /home/azmudray/ns-allinone-2.35/ns-2.35/
$] make
```

Once the compilation was done, the jellyfish behavior was implemented with jellyfish Tcl and numbers of malicious nodes were declared. Figure 1 shows how jellyfish is declared in Tcl file and recompiled before network animation (NAM) was used to simulate it against the protocols. The declaration of jellyfish attacks has to do with the position and time at which jellyfish attack needed to start to be effective in the simulation. In the above declaration the four nodes 5, 8, 10, 18 were initially declared as malicious node at time 0.0 except node 18 which was declared after 10sec. This implies that node 18 starts to act as malicious at 10 sec in simulation. Node 5 was deactivated by adding comment sign at the beginning of its declaration.

Figure 2 shows how jellyfish is declared in Tcl file and recompiled before network animation (NAM) was used to simulate it against the protocols. In the above declaration the node 8 is active while the other nodes were commented to deactivate them. Which means that a single node is enable act as malicious, this is only node that will drop packet.

Figure 3 shows how jellyfish drop packet in animation (NAM) mode during simulation. Packets received by the malicious node are drop instead of forwarding the packet to the neighbouring nodes.

2.6 *Performance Evaluation*

2.6.1 *Quantitative Metrics*

The simulation measurements used are view in two ways i.e. either externally or internally. The external measurements are throughput, end-to-end delay, and packet delivery ratio while the internal behaviors are subdivided into two:

Routing Efficiency: how much of the sent data is delivered to the destination in the present of the attacks.

Routing Accuracy: Hops count compared to the optimal shorted path.

2.6.2 *Parameters*

The metrics has been measured against some parameters that described the characteristic behavior of ad-hoc network and can be varied in a controlled way.

The parameters that are considered to simulate in this research are stated below:

- Mobility: affect the dynamic topology; link will go up and down
- Offered Network load: packet size, number of connections and the rate that packet is being sent
- Network size: number of nodes, the size of the area that the nodes are moving within. Fewer nodes in the same area mean fewer neighbors to send requests to.
- No of malicious nodes: this varies in size, more attacks are introduced, and effect of the attacks on the protocols was measured.

2.7 *Performance Metric*

The listed performance metrics are considered for evaluation of the protocols in the presence of jellyfish attacks:

- Packet Delivery Ratio (PDR): it is the ratio of packet received by the destination to the packet originated from source.

$$PDR = Pr/Ps$$

Where Pr is total packet received and Ps is the total packet sent

- Average end-to-end delay: it is defined as the time taken for a data packet to be transmitted across

MANET from source to destination.

$$D = (Tr - Ts)$$, where Tr is receive Time and Ts is sent Time

Average Throughput: This metrics represents the average number of bits arrived per second at destination and measured in bps.

2.8 *Result and Discussion*

2.8.1 *Performance Analysis*

The result of the simulation was generated at maximum speed of 200 m/s, 25 nodes, 500 meters x 500 meters, simulation time 200 sec, pause time 10 sec and at the varying number of attacks from 1-5.

A. *Average Throughput*

Table 3 shows the result of Average throughput of AODV and RAODV under jellyfish attacks. This part of analyzing is important as it describes the average number of bits arrived per second at destination and measured in bps. Table 3 shows that the average throughput of AODV and RAODV started from 51.08kbps while the malicious node is 1 and as it increases to 2, average throughput decreases to 50.00kbps while at malicious node 3 and 4 average throughput remains constant i.e. 49.89kbp and later decreases to 31.13kbps at 4 malicious nodes. This implies that the average number of bits arrived per second at destination decreases with increase in number of malicious nodes.

B. *Packet Delivery Fraction*

Table 4 shows the result of Packet Delivery Fraction (PDF). This part of analysis is important as it describes the rate of packets drop as well as it affects the overall network throughput that the network can support. Table 4 shows that the packets delivery ratio is 99% for AODV and RAODV when malicious node is 1. When malicious nodes increase to 2 the packet delivery fraction reduces to 58%. When malicious node increases to 3 the packet delivery fraction remain constant. This implies that the packet delivery rate decreases as number of jellyfish attack increases.

C. *Packet Drop*

Table 5 presents data packets drop. Reactive routing protocol generates or flood packets on demand only, in order to reduce routing loads. This proactive periodically message exchanging, causes tremendous routing overhead. This minimizing of information exchange for reactive routing protocol is an advantage for reactive routing protocol and will result in minimizing dropped data packets. In the presence of jellyfish periodic dropping, packet drop increases. Packet Drop = Packet sent - packet received. From the chart at the first node packet drop to 8 while increases to 521 at 2 nodes, 3 and 4 nodes with constant drop of 523packets this later decreases to 490. Table 5 shows that multiple jellyfish attacks have a greater packet drop effect on AODV and RAODV than a single attack.

D. *End to End Delay*

Table 6 shows End to End Delay and Figure 6 illustrates end to end delay result for AODV and RAODV.

The Figure 1.5 shows that the delay of 25nodes varies depend on the available attacks on a network. The end-to-end was calculated as Tr-Ts. Where Ts is Time sent and Tr is Time received. The value starts from 98.96% and decreases down to 60.03% and rises back to 98.96%s. This means that end-to-end decreases as jellyfish attacks increase in AODV and RAODV.

2.8.2 *Performance Metrics Calculation*

- Packet Delivery Ratio (PDR): = Pr/Ps
  Where Pr is total packet received and Ps is the total packet sent
  Pr = 4145, Ps = 6210
  PDR = 4145/6210 = 0.667
  Table 3.5 shows Packet Sent and Packet Received
- Average end-to-end delay: it is defined as the time taken for a data packet to be transmitted across MANET from source to destination. Table 3.6 shows Sent and Received Time

D = (Tr – Ts), where Tr is receive Time and Ts is sent Time

3. *Conclusion and Future Work*

This research was carried out with pre-defined parameters which are pause time, simulation time, numbers of nodes and protocols with several experiments and analysis. The effects of malicious nodes were varying to evaluate the performance of AODV and RAODV protocol. The simulation results evaluated show the performance of the routing protocol with regard to four performance metrics i.e. Packet Delivery Fraction, Average Throughput, Average End-to-End delay, Packet Drop. The experimental results for all scenarios show that Packet Delivery Fraction of AODV and RAODV at first jellyfish attack was high while it decreases nearly to half of the first in 2 attacks then it continues to decrease with little change in values. From the results, it was gathered that average number of bits arrived per second at destination decreases with increase in number of jellyfish nodes.

At the point of 3 and 4 attacks, the average throughput of the two protocols remains constant while it drastically decreases when the 5th attacks was introduced. The time taken for a data packet to be transmitted across MANET from source to destination started at peak level and decreases from 2nd to 4th malicious nodes but returned back to initial value at node 5. This means the end-to-end delay of the two protocols gradually decreases and later increases at the last test. The results also shown that the number of packet drop at first attack is lesser compare to when the number of threat increases, this means that at the presence of multiple threats much packets dropped and less at the first threat.

From the analysis, it could be concluded that the higher the jellyfish attacks on AODV and Reverse -AODV, the lower the performance of the protocols when the parameters like pause time, maximum speed, number of nodes are kept constant. Moreover, further research may be carried out with the use of parameters such as normalized routing overhead, network mobility diverge numbers of node to calculate the performance metrics of the two protocols.

References

Akanksha S. and Kumar (2010) "Effect of Black Hole Attack On AODV Routing Protocol In MANET". International Journal of Computer and Telecommunication. Vol. 1, Issue 2, pg. 148.

Chetana K. (2013) "Black Hole Node Detection in AODV" International Journal of Computational Engineering Research. Vol, 03, Issue, 6, pg 79.

Esmaili et al (2011) "Performance Analysis of AODV under Black Hole Attack through Use of OPNET Simulator". Computer Science and Information Technology Journal Vol.1, No.2, pg 49-52.

Hetal P. , Minubhai. B. (2010) "Survey: Impact of Jellyfish On Wireless Ad-Hoc Network", in proceeding of INJCR'10, Volume.10, issue.5, no.2 pg. 5-9.

Kamanshis B. and Liakat A. (2007) "Security Threats in Mobile Ad Hoc Network". Interaction and System Design, School of Engineering, Blekinge Institute of Technology.

Pratibha Kaswan and Deepika Gupta (2013) "Impact Analysis of Wormhole and Black hole attacks over Mobile Ad Hoc Networks". Computer Science and Information Technology Journal. Volume: 1 Issue: 12, pg 8.

Rehman, I. (2010) "Analysis of Black Hole Attack on MANETs, Using Different MANET Routing Protocols", School of Computing, Bleking Institute of Technology.

Sanabani M. et al (2014) "A Reverse and Enhanced AODV routing protocol for MANETS". ARPN Journal of Engineering and Applied Sciences. Vol 9, No 2, pg 11.

Tanwar S., Prema K.V. (2013) "Threats & Security Issues in Ad hoc network: A Survey Report". International Journal of Soft Computing and Engineering (IJSCE), Volume-2, Issue-6, pg 1-6.

Table 1: Simulation Parameters

Parameter	Value
Simulator	Ns-2.35
Data packet size	512byte
Simulation time	200sec
Total number of nodes	25
Data rate	11Mb
Pause time	10s
Observation parameters	PDR, Throughput, end-to-end delay, packet drop
No of malicious node	1-5
Routing protocols	RAODV and AODV
Simulation Size	500m x 500m

Table 2: Simulation Setup

Constant	Vary
Number of Nodes 25 with Pause Time(10s)	Malicious nodes(1)
Number of Nodes 25 with Pause Time(10s)	Malicious nodes(2)
Number of Nodes 25 with Pause Time(10s)	Malicious nodes(3)
Number of Nodes 25 with Pause Time(10s)	Malicious nodes(3)

Table 3: Average throughput

MaliciousNode	AvgThroughput	StartTime	StopTime
1	51.08	1	99.96
2	50	1	60.06
3	49.89	1	60.03
4	49.89	1	60.03
5	31.13	1	99.96
6	0	1	0

Table 4: Packet Delivery Fraction

Malicious Node	Sent	Received	r/s	% r/s	F
1	1242	1234	0.9936	99%	3320
2	1242	721	0.5805	58%	3901
3	1242	719	0.5789	57%	2885
4	1242	719	0.5789	57%	2885
5	1242	752	0.6055	60%	2358
6	485	0	0	0	226

Table 5: Packet Drop

Malicious Node	Sent	Received	Packet Loss
1	1242	1234	8
2	1242	721	521
3	1242	719	523
4	1242	719	523
5	1242	752	490
6	485	0	485

Table 6: End to End Delay

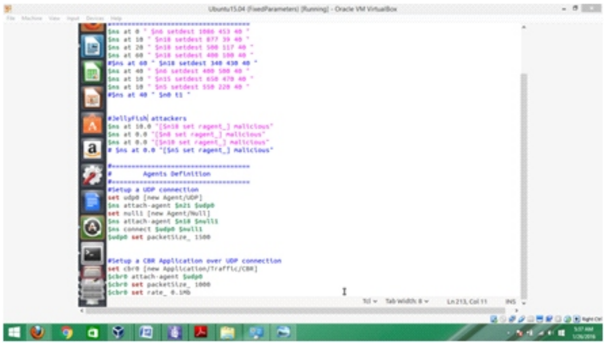
Malicious Node	AvgThroughput	StartTime	StopTime	AvgEnd-to-End
1	51.08	1	99.96	98.96
2	50	1	60.06	59.06
3	49.89	1	60.03	59.03
4	49.89	1	60.03	59.03
5	31.13	1	99.96	98.96

Table 7: Packet Sent and Packet Received

Malicious Nodes	Packet Sent(s)	Packet Received(s)
1	1242	1234
2	1242	721
3	1242	719
4	1242	719
5	1242	752
Total	**Ps = 6210**	**Pr = 4145**

Table 8: Sent and Received Time

Malicious Nodes	Sent Time(s)	Received Time(s)
1	1.00	99.96
2	1.00	60.06
3	1.00	60.03
4	1.00	60.03
5	1.00	99.96


Ubuntu15.04 (FixedParameters) [Running] - Oracle VM VirtualBox

```
#==================================
$ns at 0 " $n6 setdest 1000 453 40 "
$ns at 10 " $n10 setdest 877 39 40 "
$ns at 20 " $n18 setdest 500 517 40 "
$ns at 60 " $n18 setdest 400 100 40 "
#$ns at 60 " $n18 setdest 340 430 40 "
$ns at 40 " $n6 setdest 400 500 40 "
$ns at 10 " $n15 setdest 650 470 40 "
$ns at 10 " $n5 setdest 550 220 40 "
#$ns at 40 " $n0 t1 "


#JellyFish attackers
$ns at 10.0 "[$n18 set ragent_] malicious"
$ns at 0.0 "[$n8 set ragent_] malicious"
$ns at 0.0 "[$n10 set ragent_] malicious"
# $ns at 0.0 "[$n5 set ragent_] malicious"

#==================================
#        Agents Definition
#==================================
#Setup a UDP connection
set udp0 [new Agent/UDP]
$ns attach-agent $n21 $udp0
set null1 [new Agent/Null]
$ns attach-agent $n18 $null1
$ns connect $udp0 $null1
$udp0 set packetSize_ 1500


#Setup a CBR Application over UDP connection
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $udp0
$cbr0 set packetSize_ 1000
$cbr0 set rate_ 0.1Mb
```

Figure 1: jellyfish is declared in Tcl file

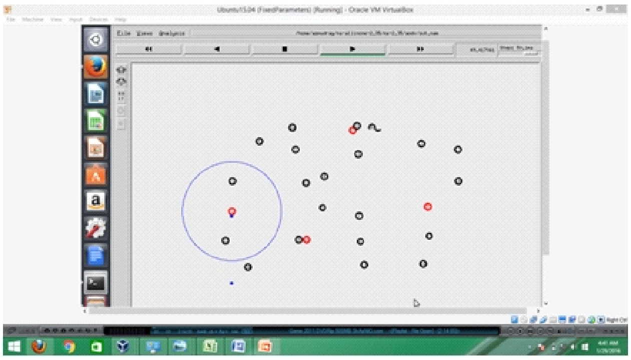

Ubuntu15.04 (FixedParameters) [Running] - Oracle VM VirtualBox
File Views Analysis

Figure 2: NAM shows jellyfish node dropping packets

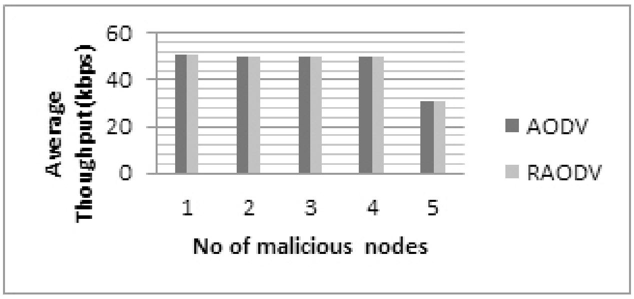


Figure 3: Average Throughput

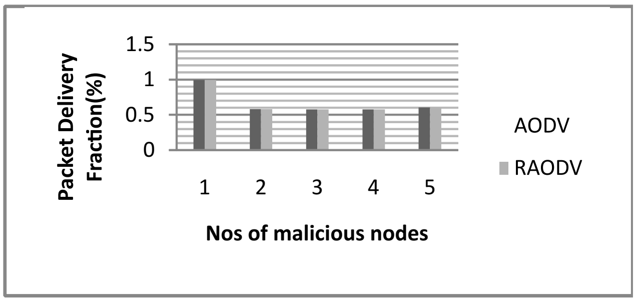


Figure 4: Packet Delivery Fraction

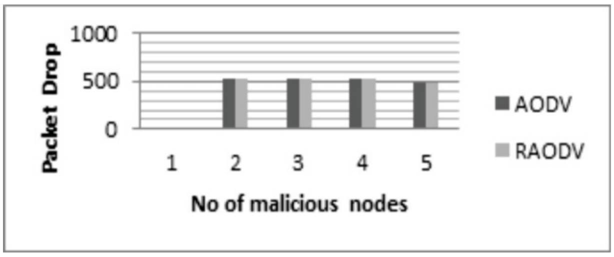


Figure 5: Packet Drop

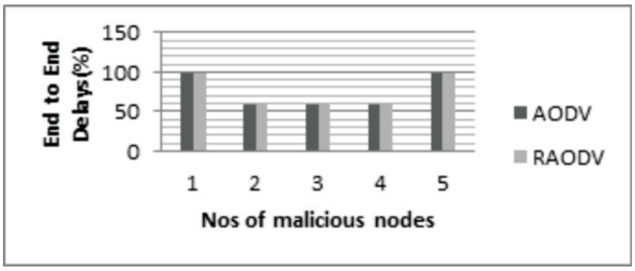


Figure 6: End to End Delay