Fuzzy Systems and Soft Computing
ISSN : 1819-4362

NAODV_ETCP Implementation to Address Jellyfish Attack

DR. GURU PRASAD DASH[1], PROF. KAMALAKANTA PADHI[2]

[1, 2] DEPT. OF COMP. SC. & ENGG.

[1]guruprasadadsh@rec.ac.in , [2]kamalakantapadhi@rec.ac.in

[1, 2] RAAJDHANI ENGINEERING COLLEGE

Abstract

Jellyfish attack is of three types namely jelly fish packet dropping attack, jelly fish delay variance attack and the jellyfish packet reordering attack. Each type of jellyfish attack degrades the performance of the network by dropping or changing sequence of the packet or by delaying the acknowledgement. The existing algorithm i.e. ETCP can handle the jellyfish delay variance attack. This paper modifies the ETCP to develop NAODV_ETCP that can handle all the three types of jellyfish attack. In the ETCP protocol the buffer stores the sequence number and the acknowledgment time while in the NAODV_TCP protocol the fr(forwarding ratio) is also stored in the buffer. This paper analyzes the performance using PDR, E2Edelay and the throughput on the various scenario attacked by different types of jellyfish attack. The result analysis shows that the performance of NAODV_ETCP is better than the ETCP protocol.

Keywords*: ETCP, AODV, NAODV_ETCP, TCP, MANET.*

1. Introduction

MANET is a self-organizing system, consisting of mobile nodes which communicate with each other through wireless links. The nodes with in MANET act as the router or forwarder that forwards the data intelligently. The MANET can be popular due to its various applications in military, emergency rescue etc. The delivery of message in the MANET can be unicast, broadcast or multicast simultaneously [1].

Mobile Adhoc network connects various mobile nodes via wireless links. The routing of data from source nodes to the destination nodes uses the link layer and network layer protocols. The connectivity can be single or multi hop but the main optimization of the routing needs guaranteed delivery of the packet to the destination [1]. The performance of any mobile adhoc network is the main issue of concern Due to its dynamic nature and high speed mobile nodes. Moreover, various network layer attack degrades the performance of the network. The attacks can be divided mainly in two categories one is data traffic attack and other is control traffic attack [2]. The data traffic attack modify the data or share the data with the third i.e. unauthorized entity. The control traffic attack the control signals that results in increase in delay or packet drop etc. Jellyfish is a control traffic attack. The paper describes an algorithm to handle the jellyfish attack.

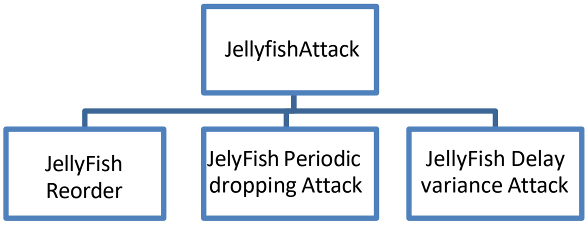

JellyfishAttack
JellyFish Reorder
JelyFish Periodic dropping Attack
JellyFish Delay variance Attack

Figure 1: Types of Jelly-fish Attack

It is a DOS attack and difficult to detect due to its passive nature. It is of three types as shown in figure 1 i.e. jellyfish packet dropping attack, jellyfish delay variance attack, jellyfish packet reordering attack. In the jellyfish packet dropping attack, the packets are dropped by the compromised nodes [3]. The packet dropping occur due to the delay in forwarding the node data. In the delay variance attack, delay is introduced before the transmission as well as reception of the data. The jellyfish attack affect the performance of the all routing protocol. It is a passive attack as it follows all protocol rules [4].

2. Related Work

Dyer T.D. et al. [6] analyze the performance of TCP over three routing protocols i.e. AODV, DSR and the ADV (adaptive proactive). The author also evaluated the performances of described algorithms on TCP Reno & Reno with fixed RTO. Aad I. et al. [7] studied the jelly-fish attack and its affect on the network. Dokurer S. et al. [8] determined the performance of AODV protocol under blackhole attack by using the NS2. The author had taken different scenarios having different number of blackhole nodes. Lou W. et al. [9] Proposed a scheme that delivers different segment of message through the different path available in the network to enhance the security.

Parmar J.D. et al. [10] studied various routing protocols and the major security concern issues in the MANET. Roopak M. et al. [11] compared the performance of AODV in different scenario (with and without attack) by using parameters like PDF, E2Edelay and throughput. Khurshid A. et al. [12] modify the TCP sender algorithm to differ the congestion and the non-congestion events by analyzing the time difference. This leads the improved performance of TCP. Mulert J.V et al. [13] proposed SAODV that includes the features like multipath routing, packet leashes and randomized route request to handle the black hole, resource depletion , worm hole etc attacks. Begum S.A. et al.

[14] discuss techniques to make protocol robust against the DoS attack. The author describes the data traffic as well as control traffic attacks and described technique is robust against such attacks. Saetang W. et al. [15] proposed CAODV i.e. AODV with credit mechanism. The CAODV protocol modify the AODV routing protocol by introducing a phenomena (credit phenomena) to detect & protect from the malicious node. Wazid M. et al. [5] designed E-TCP for reducing the congestion. The performance enhancement is analyzed in the E-TCP. It can handle the delay variance jelly-fish attack. The detail of this protocol is discussed in next section.

3. E-TCP Protocol

The Efficient TCP protocol is the modified TCP protocol proposed by the author of [5]. TCP is reliable protocol that transmit the acknowledgement upon receiving the packet [5]. The jellyfish attack degrades the performance of the network. The ETCP disable the fast transmission and enables the selective acknowledgement to control the congestion situation in case of the jellyfish attack. The parameters of TCP Protocols are modified to handle the jellyfish attack. The working can be understood by following algorithm:

```
**ETCP ()**
Start
// pda_buf [seq] [time] is a two dimensional array having two columns used as buffer.
 Seq is the sequence number, time is the ack time.
P is the processing time.
CH is the cluster head node
T is the time delay at each node.
l is the length of route.
{
Transmit the packet from source to destination on selected path
For each Node say N
        For i=1:n
            Seq=i;
          For j=0:l-1
              If N=Source
                   Time=time+lT
              else
                 time=time+lT+$P_N$
              end if
             Insert data to pda_buf.
         End for
End for
For each entry in pda_buf of CH
 If (pda_buf entry at CH=pda_buf entry at intermediate node) then
   ETCP ();
 Else
     Disable fast_retransmission;
     Enable selective_ACK;
 End
End
stop
}
```

The algorithm can handle only the delay variance jellyfish attack. The next section describe the present work to handle all types of jellyfish attack.

4. Proposed System

The proposed system i.e. NAODV_ETCP modifies the existing system i.e. AODV_ETCP to handle the jelly fish periodic packet dropping attack, the jelly fish delay variance attack as well as the jelly fish reorder attack. The proposed system uses the E_TCP of the existing system along with the modified AODV routing to get the effective results. The NAODV_ETCP process handle the packet dropping attack as follow:

1. By Checking the forwarding Ratio at each node.
2. By sending ACK from each node instead of destination.

The source node broadcast the RREQ message and the group of nodes at one hop distance receives the request. The nodes with forwarding ratio less than the threshold value gets discarded from the group. The forwarding ratio is calculated by number of packet received divided by number of packet forwarded. The node with forwarding rate less than 0.70 i.e. 70% is discarded i.e. the threshold is 0.70. The remaining nodes of the group receive the packet and send the acknowledgement. The process continues until destination reached. If any node receive the acknowledgement from the destination but not from the neighbor node then the node discard the neighbor node. This process handles the packet dropping attack. The NAODV_ETCP handle the reordering attack by using a buffer at each node which receives the request and forward it only after reordering. The length of buffer is 5 i.e and the buffer forward the request when it is full by 60%. The jellyfish delay attack is handled by calculating the average delay time of the routing path. If the packet doesn't reach the destination within th+average delay time than the packet is discarded and the route is marked as the non-usable route; where th is threshold value i.e. constant value for any particular network. The other packet transmission doesn't prefer the route.

The whole process can be easily understood by the following algorithm:

1. The Source node say S and the destination node say D is selected.
2. The S node transmit the hello packet.
3. Ad=the time taken by hello packet to reach the destination.
4. T=0;
5. Select cur=S
6. First_t=0;
7. While cur ~=D
8. Broadcast the RREQ from cur after reordering at cur.
9. G=Group of nodes at one hop distance from cur.
10. If first_t==0
11. First_t=1
12. else
13. If the cur receives the Ack from destination but not from neighbor
14. Then discard the neighbor node
15. End if
16. End if
17. For each node in G say Ni
18. If forwarding ratio of node Ni<0.70
19. Then discard the node
20. End if
21. Store the RREQ in the buffer of Ni.
22. Send Ack from each node Ni.
23. End for
24. Update cur.
25. t=t+current_time_taken
26. If t>ad+th
27. Then discard the path.
28. cur=S
29. End if
30. End while

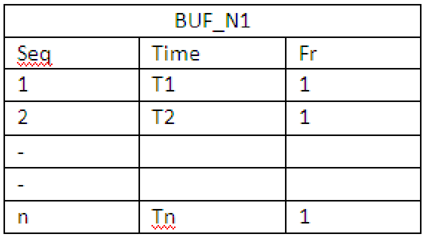
The pseudo code implementation of the above steps is given below:

NAODV_ETCP()

```
Start
// pda_buf[seq] [time] [fr] is a two dimensional array having three columns used as buffer to handle the packet dropping and delay variance attack.
Seq is the sequence number, time is the ack time and fr is the forwarding ratio.
P is the processing time.
CH is the cluster head node
T is the time delay at each node.
l is the length of route.
{
Transmit the packet from source to destination on selected path
For each Node say N
        For i=1:n
            seq=i;
          For j=0:l-1
              If N=Source
                  time=time+lT
                   fr=1;
              else if delay attack at N
                  time=time+lT+$P_N$+del
                  fr=fr-1
              else if packet dropping attack
                   time=inf;
                   fr=fr-1;
              else
                  time=time+lT+$P_N$
                    fr=fr+1
              end if
              Insert data to pda_buf and sort by seq.
        End for
End for
for each entry in pda_buf of CH
  if (pda_buf entry at CH=pda_buf entry at intermediate node) and fr>0.7 then
    NAODV_TCP ();
  else
     Disable fast_retransmission;
     Enable selective_ACK;
  End
End
stop
}
```

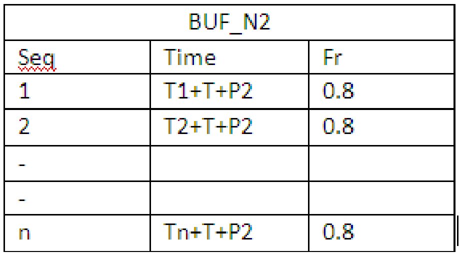
The proposed algorithm is an efficient algorithm i.e. used is capable to handle the jelly-fish attack of all types. It can also be understood by following example:

The buffer at different nodes are shown in following figures. Here, the seq represents the sequence number and the time includes the processing as well as the channel transmission time. The channel transmission time is T and the processing time for nodes are P2, P3……Pl where l is the length of route.



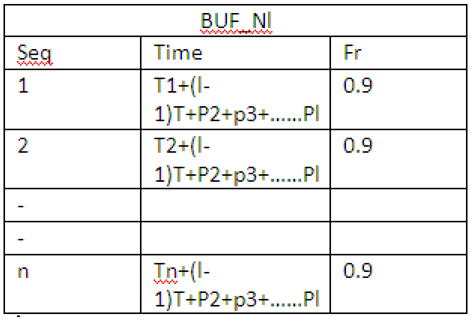
BUF_N1		
Seq	Time	Fr
1	T1	1
2	T2	1
-		
-		
n	Tn	1

Figure 2: Buffer at node 1



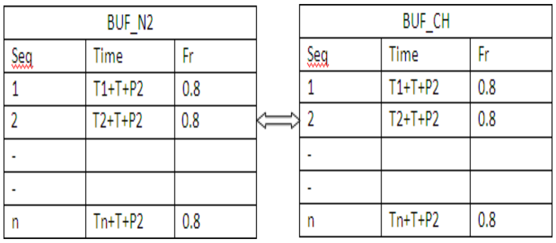
BUF_N2		
Seq	Time	Fr
1	T1+T+P2	0.8
2	T2+T+P2	0.8
-		
-		
n	Tn+T+P2	0.8

Figure 2: Buffer at node 2



BUF_Nl		
Seq	Time	Fr
1	T1+(l-1)T+P2+p3+......Pl	0.9
2	T2+(l-1)T+P2+p3+......Pl	0.9
-		
-		
n	Tn+(l-1)T+P2+p3+......Pl	0.9

Figure 3: Buffer at l node

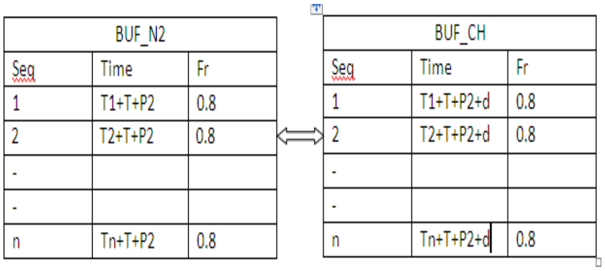


BUF_N2		
Seq	Time	Fr
1	T1+T+P2	0.8
2	T2+T+P2	0.8
-		
-		
n	Tn+T+P2	0.8

BUF_CH		
Seq	Time	Fr
1	T1+T+P2	0.8
2	T2+T+P2	0.8
-		
-		
n	Tn+T+P2	0.8

Figure 4: Normal Flow

The figure 4 shows the normal flow as the time matches at N2 and CH.



BUF_N2		
Seq	Time	Fr
1	T1+T+P2	0.8
2	T2+T+P2	0.8
-		
-		
n	Tn+T+P2	0.8

BUF_CH		
Seq	Time	Fr
1	T1+T+P2+d	0.8
2	T2+T+P2+d	0.8
-		
-		
n	Tn+T+P2+d	0.8

Figure 5: Attacked Flow

The figure 5 shows the mismatch of data resulting the attack identified at node 2.The implementation and result analysis of this algorithm is done in the next section.

5. Simulation Results

The implementation and result analysis of this algorithm is done by using the simulator NS2. The proposed technique is implemented in NS-2.35 Simulator in Linux environment. The tcl file is executed and it generates a .nam file which can be viewed in Network Animator tool of ns2 simulator.

5.1 Performance Metrics

Following are the metrics from which we calculate the performance of the network:

- **Throughput**

Throughput or network throughput is the average rate of successful message delivery over a communication channel. This data may be delivered over a physical or logical link, or pass through a certain network node. The throughput is usually measured in bits per second (bit/s or bps), and sometimes in data packets per second or data packets per time slot.

- **Packet Delivery Ratio (PDR)**

The ratio of the number of delivered data packet to the destination. This illustrates the level of delivered data to the destination.

∑ Number of packet receive / ∑ Number of packet send

- **End-to-End Delay**

The average time taken by a data packet to arrive in the destination. It also includes the delay caused by route discovery process and the queue in data packet transmission. Only the data packets that successfully delivered to destinations that counted.

∑ (arrive time – send time) / ∑ Number of connections

5.2 Results

The comparison of the existing ETCP and the proposed NAODV_ETCP protocol is done under all the three types of jelly fish attacks. The comparison is done in normal scenario i.e. under 0 attacked nodes and in scenario having 1 and 3 attacked node. The fig 1 & fig 2 and 3 shows that the comparisons under the packet dropping attack. The fig 2 shows the comparison of the PDR. While the fig 3 compares the e2delay in ms and the fig 4 compares the throughput. The throughput is measured in kbps. The fig 6,7,8 clearly shows the better performance of the proposed system as compared to the existing system.

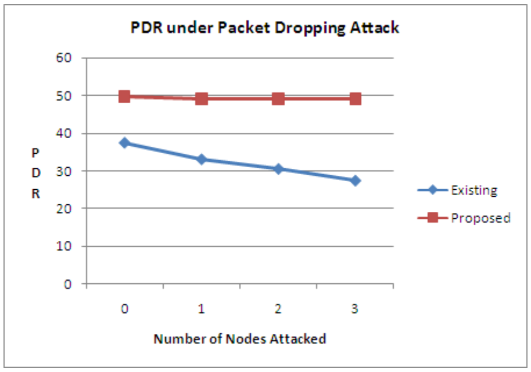


Figure 6: PDR Analysis under Packet Dropping Attack

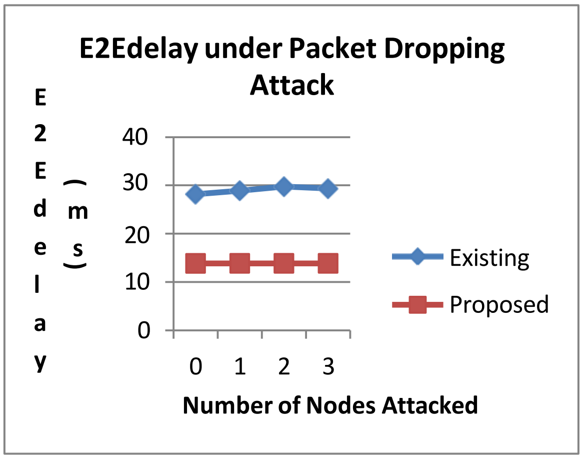


Figure 7: E2Edelay analysis under packet dropping attack

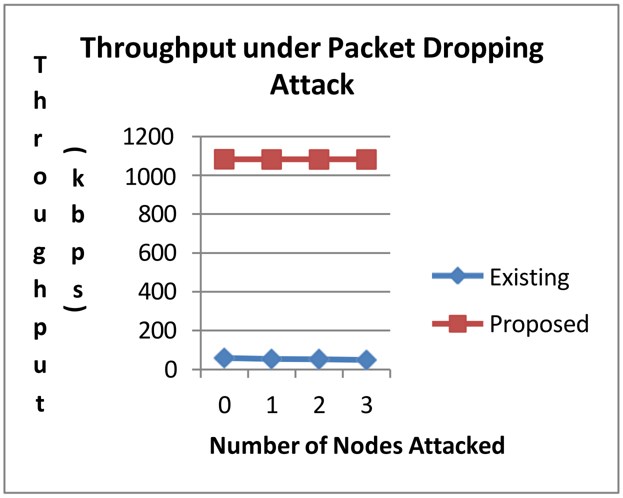


Figure 8: Throughput analysis under packet dropping attack

The fig 9, 10 and 11 shows the comparison of the proposed and the existing system under the delay variance attack. The fig 9 compares the PDR and the enhancement in the PDR of the proposed system as compared to existing algorithm confirms the better performance of the proposed protocol.

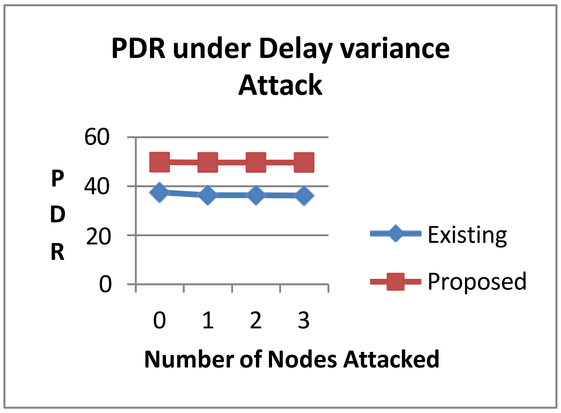


Fig 9: PDR Analysis under Delay Variance Attack

The fig 10 compares the e2edelay in the presence of the delay variance attack. In the normal scenario as well as in attacked scenario the performance of the proposed is better than the existing protocol.

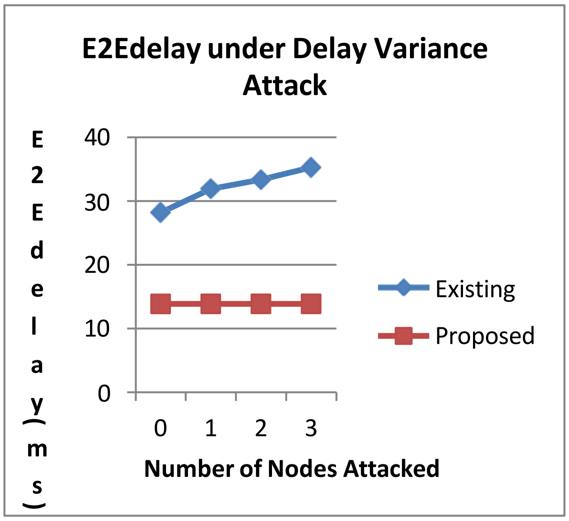


Figure 10: E2Edelay analysis under delay Variance attack

The fig 11 compares the throughput and enhancement in the throughput can be analyzed. The fig 12, 13 and 14 compares the parameters under the packet reordering jellyfish attack. The fig 12 compares the PDR and the improvement in the PDR can be recognized.

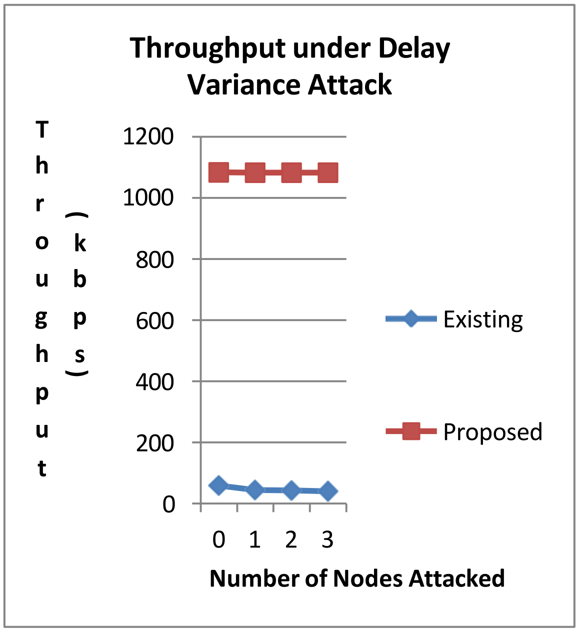


Figure 11: Throughput analysis under delay Variance attack

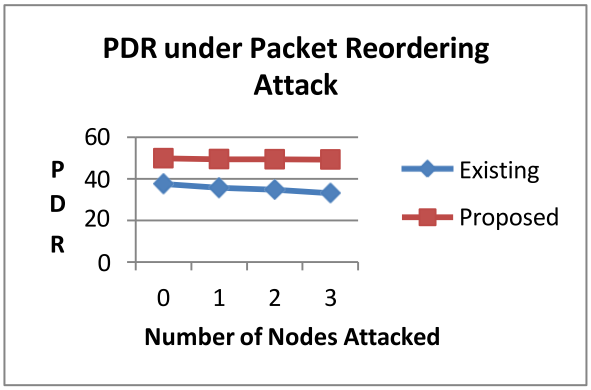


Figure 12: PDR Analysis under Packet Reordering Attack

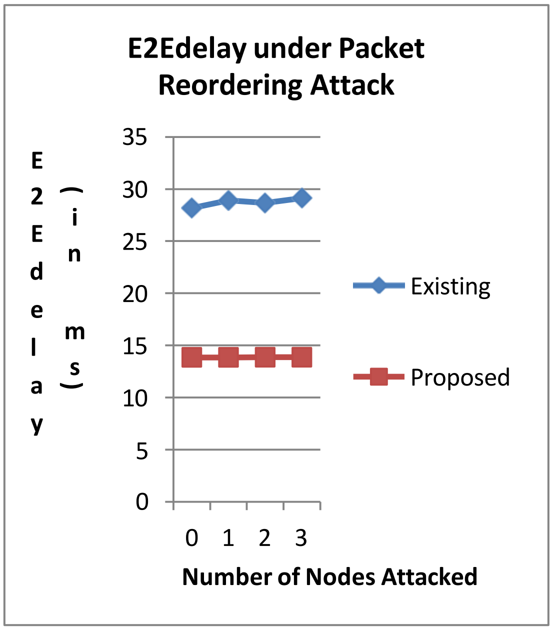


Figure 13: E2E Delay Analysis under Packet Reordering Attack

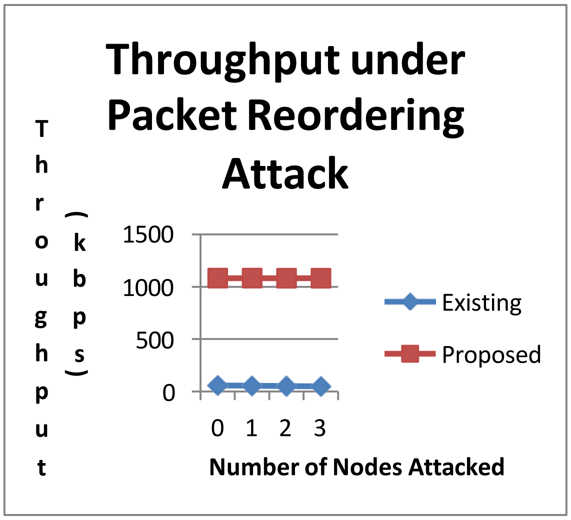


Figure 14: Throughput Analysis under Packet Reordering Attack

The graphical and the tabular analysis show that the performance of the proposed protocol is better than the existing protocol. The performance analysis shows that the NAODV_TCP is able to handle all type of jellyfish attack. The graphical and the tabular analysis show that the performance of the proposed protocol is better than the existing protocol. The performance analysis shows that the NAODV_TCP is able to handle all type of jellyfish attack.

6. Conclusion

The paper modifies the ETCP protocol to develop the NAODV_ETCP protocol. The ETCP protocol can handle only one type of jelly fish Attack while the NAODV_ETCP protocol can handle all the three types of jellyfish attack. The protocol is analyzed against the ETCP protocol under each attack and the performance of the ETCP protocol is better than the NAODV_ETCP protocol in all the cases. The PDR is improved with the decrease in e2edelay. This shows the improved performance of the system. Moreover the throughput of the NAODV_ETCP is better than the ETCP protocol. This proves the better QoS of the NAODV_ETCP protocol as compared to the ETCP protocol. In future the work can be extended to handle other types of attacks possible in the network. The meta-heuristic techniques can also be used to enhance the performance of the designed protocol.

References

[1] Nguyen, Hoang Lan, and Uyen Trang Nguyen. (2008),A Study Of Different Types Of Attacks On Multicast In Mobile Ad Hoc Networks., Ad Hoc Networks 6, no. 1.

[2] Bhattacharyya, Aniruddha, Arnab Banerjee, Dipayan Bose, Himadri Nath Saha, and Debika Bhattacharya. (2011) Different types of attacks in Mobile ADHOC Network., arXiv preprint arXiv:1111.4090.

[3] Amandeep Kaur et al (2013) Effects of Jelly Fish Attack on Mobile Ad-Hoc Network's Routing Protocols, Int. Journal of Engineering Research and Applications www.ijera.com ISSN : 2248-9622, Vol. 3, Issue 5, Sep-Oct 2013, pp.1694-1700

[4] Mohammad Wazid, Vipin Kumar, RH Goudar, "Comparative performance analysis of routing protocols in mobile ad-hoc network under Jelly fish attack", 2nd IEEE International Conference on parallel, distributed and grid computing, 2012

[5] Wazid, M., Katal, A., Sachan, R. S., & Goudar, R. H. (2013, April). E-TCP for efficient performance of MANET under JF delay variance attack. In Information & Communication Technologies (ICT), 2013 IEEE Conference on (pp. 145-150). IEEE.

[6] Dyer, Thomas D., and Rajendra V. Boppana.(2001) , A comparison of TCP performance over three routing protocols for mobile ad hoc networks., In Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing, pp. 56-66. ACM,.

[7] Aad, Imad, Jean-Pierre Hubaux, and Edward W. Knightly. (2004),Denial of service resilience in ad hoc networks., In Proceedings of the 10th annual international conference on Mobile computing and networking, pp. 202-215. ACM,.

[8] Dokurer, Semih, Y. M. Erten, and Can Erkin Acar. .(2007) Performance analysis of ad-hoc networks under black hole attacks., In SoutheastCon, 2007. Proceedings. IEEE, pp. 148-153. IEEE, 2007.

[9] Lou, Wenjing, Wei Liu, Yanchao Zhang, and Yuguang Fang. (2009), SPREAD: Improving network security by multipath routing in mobile ad hoc networks. Wireless Networks 15, no. 3:

[10] Jhaveri, Rutvij H., Ashish D. Patel, Jatin D. Parmar, and Bhavin I. Shah. (2010) MANET routing protocols and wormhole attack against AODV. International Journal of Computer Science and Network Security 10, no. 4: 12-18.

[11] Roopak, Monika, and Dr Bvr Reddy. (2011): Performance Analysis of Aodv Protocol under Black Hole Attack. International Journal of Scientific & Engineering Research 2, no. 8 1.

[12] Khurshid, Ahmed, Md Humayun Kabir, and Md Anindya Tahsin Prodhan. (2011),An improved TCP congestion control algorithm for wireless networks. In Communications, Computers and Signal Processing (PacRim), 2011 IEEE Pacific Rim Conference on, pp. 382-387. IEEE,.

[13] Von Mulert, Jan, Ian Welch, and Winston KG Seah. (2012). Security threats and solutions in MANETs: A case study using AODV and SAODV. Journal of Network and Computer Applications 35, no. 4.

[14] Begum, Syed Atiya, L. Mohan, and B. Ranjitha. (2012).Techniques for Resilience of Denial of Service Attacks in Mobile Ad Hoc Networks., Proceedings published by International Journal of Electronics Communication and Computer Engineering 3, no. 1.

[15] Saetang, Watchara, and Sakuna Charoenpanyasak. (2012)., CAODV Free Blackhole Attack in Ad Hoc Networks. International Proceedings of Computer Science & Information Technology 35.