DETECTION AND PREVENTION TECHNIQUE FROM JELLYFISH DELAY VARIANCE ATTACK

Hardik Prajapati[1], Ashish Patel[2], Kunjal Brahmbhatt[3]
[1,3]*Indus Institute of Technology*
[2]*Silver oak Institute of Technology*

***Abstract*- An ad hoc network is an accumulation of wireless mobile nodes dynamically forming a tentative network without the use of any existing network infrastructure or medication administration. Due to the limited transmission range of wireless network interfaces, multiple network "hops" may be needed for one node to transposition of data with another node across the network. In recent years, wireless ad hoc networks (WANETs) have become very in-vogue due to their wide range of petition and their ability to be deployed under normal and rugged conditions while supporting high data rates. Although many intrusion detection and trust-based systems have been developed to protect ad hoc networks against misconduct such as rushing attacks, query-flood attacks, and selfishness of nodes, these aegis mechanisms are still not able to detect protocol compliant attacks called *Jellyfish* (JF) attacks. They target *closed-loop* flows such as TCP that are responsive to network conditions like delay and packet losses and can easily partition the network. In this paper, we introduce a technique which can be used to detect and detain Jellyfish delay variance attacks in ad hoc networks**

***Index Terms*- AODV, end to end delay, Jellyfish Attacks, Mobile Ad-hoc Network.**

I. INTRODUCTION

Emergent progress has been made in securing ad hoc networks by developing secure routing protocols that certify different security concepts such as authentication and data integrity. Moreover, intrusion detection and trust-based systems have been developed to defend WANETs against misbehavior such as rushing attack, query-flood attacks, and selfish behaviors. Yet, most of the salvage mechanisms are not able to detect a set of protocol compliant attacks called *jellyfish* (JF) attacks.

Similar to a jellyfish which is difficult to be detected until after the sting, jellyfish attacks in ad hoc networks are also hard to detect because they conform to all existing protocol specifications. Jellyfish attackers (JF nodes) specially delay variance attacker could not change or modified the data, but only make alteration in such a way that TCP's feature invoke unnecessarily such that performance of network start degrade. This attack is not affected in the network which is used UDP protocol because there are no any kind of feature that the UDP has to invoke unnecessarily.

II. LITERARURE REVIEW

The paper [1] discusses two protocols Ad-hoc On-demand Distance Vector (AODV) and a secure extension to AODV, the Secure AODV (SAODV) protocol. SAODV is a secure version of the AODV protocol as it relies upon the use of cryptographic mechanisms in order to protect the routing control messages of AODV from being altered by attackers. A vulnerability analysis of SAODV is done to identify unresolved threats to the algorithm, such as medium access control layer misbehavior, resource depletion, black holes, wormholes, jellyfish and rushing attacks. In paper [2] authors explain various techniques for the resilience of denial of service attacks on a mobile ad hoc network along with introducing three kinds of JellyFish (JF) attacks i.e. JF Reorder Attack, JF Periodic Dropping Attack, JF Delay Variance Attack. Throughput of network under these attacks is also calculated in this paper. Some techniques to protect MANET i.e. Flow-Based Route Access Control (FRAC), Multipath Routing Source-Initiated Flow Routing, Sequence Numbers etc. are also proposed. In [3] authors calculate the performance of MANET under black hole attack using AODV routing protocol with HTTP traffic load. In [4] authors propose an identity (ID) Based scheme that secures AODV and transmits TCP data to the authorized hosts. This scheme secures the AODV using sequential aggregate signatures (SAS) which are based on RSA and also securely generates the session key for nodes to secure the TCP. Authors on paper [5] proposed a modified TCP congestion control algorithm. In the presence of proposed algorithm the performance of the network is improved up to certain extent. In [6] an algorithm that detects the Jellyfish attack at a single node is developed which can be effectively deployed at other nodes. They also propose the novel metric that detects the Jellyfish reorder attack based on the Reorder density which is a basis for developing a metric. A comparison table is given at the end which shows the effectiveness of Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT 2013)978-1-4673-5758-6/13/$31.00 © 2013 IEEE 145 novel metric which helps protocol designers to develop the counter strategies for JF attack. In paper [7] authors present a different approach to improve the Gray Hole, Worm Hole attacks so that detection can be avoided. In paper [8] authors proposed a modified Transmission Control Protocol, TCP Reno. It is named as TCP Congestion Control Enhancement for Random Loss (CERL) which has improved the performance of TCP. To estimate the queue length of the link this protocol utilizes the RTT measurements they are made throughout the duration of the connection. In [9] the most common types of attacks on MANET, namely Rushing attack, Blackhole attack, Neighbor attack and JellyFish attack are discussed along with the simulation of these attacks. Parameters such as Average end-to-end delay, Average throughput etc. are also calculated. Authors in paper [10] also discuss about JellyFish and Blackhole attacks. Authors calculate the impact of JF on the system performance i.e. Throughput etc. Three factors: mobility, node density and system size are introduced and the effect of these factors on fairness to receive packets under the presence of various number of JF attackers is calculated. Observation made is that the effect of mobility is more under the absence of JF attackers and fairness reduces with increase in mobility. Under JF attacks the mobility is less important as all nodes suffer equivalently. Decrease in node density decreases fairness also. In paper [11] authors proposed a new version of TCP that maintains high throughput when reordering occurs. When packet reordering does not occur proposed version is friendly to other versions of TCP. In [12] the concept of efficient TCP that is E-TCP explain in which they form cluster head and that periodically measure delay of forwarding group each node and match that data to the data that it calculate previously and if present data and stored data will not matched than it will generate its mechanism to prevent this attack.

III. PROBLEM DEFINITION AND NOVELITY

TCP is a reliable protocol. Reliability ensures that once a packet is sent by source must reach destination with in time. An ACK (acknowledgment) packet is sent from destination back to source when packet sent by source reaches destination. JF delay variance attack in the network delays the packet for some amount of time before forwarding them to the destination. As a result of which ACK packet is also delayed. The source assumes that the packet has been lost so it retransmits the same packet again. This retransmission of packets occurs again and again as the packet doesn't reach destination in time due to JF delay variance attack resulting congestion in the network, which further produces delay in the network and reduces the throughput.

In this paper modification is achieved in such a manner that here we are check a network for any malicious activities before AODV initiate and start communication and we have also checked this kind of activities after a path has been selected for communication by AODV

Here in E-TCP which is Efficient TCP technique which is used for detect and prevent jellyfish delay variance attack is using centralized Entity which is cluster head (CH) but main problem is that all the control mechanism is done by Cluster head(CH) so as a attacker one must be focus on the Cluster Head(CH) .Once's is under the influence of attacker than this technique is totally fail. Secondly this method is not solving the actual problem of reducing End to End delay but it only activated selective acknowledgement mechanism such that traffic on network has reduced not End to End delay Another and serious problem is that if attacker changes information inside the cluster head (CH) match table than after cluster head is not going to detect any malicious node in the forwarding path even after malicious node is available

IV. PROPOSED DETECTION MODEL

Here detection is achieved in two phase first when network installed and secondly when AODV select its path. A reason for such assessment is to identified Malicious nodes in the network at different stages for example at beginning of the network as well as at time when communication has started.so when attacker attacks at any of phase of communication the attack easily identified over network. Here mechanism of proposed solution is stared such that any random node which is selected by the installer is initiated communication by making a check packet and forwarded to its immediate neighbors only by setting its hop count value to “1” such that when its sends to immediate neighbors than hop count become “0” from “1” so it could not forwarded further this mechanism is used to avoid unnecessary flooding of check packet here sender also save sending time of check packet so when its immediate neighbors receive that packet and process them as a normal packet and send back to sender than sender received that packet and simply do Receive time minus Send Time and check the available output with the threshold value and if this value will less than threshold than sender mark that neighbor as a Normal node and if that value will grater than threshold than sender mark that neighbor as a malicious node .Such and this process repeat by their all nodes in the network and checks whether their immediate neighbors are malicious or not and make entry n their table. So the concept is that when AODV initiated its optimum pat finding mechanism it also check availability of malicious node in the selection of the path that means if the malicious node available in the optimum selected path so AODV will not select that path even though it is shortest but AODV reject that path and select another path that do not have malicious node even though it will longer.

This is the first phase at which we detect the malicious node in our network ,than after as by describe method a path has been selected and sender starts communication with the destination,Now in second phase if attacker has not detected in first phase than our second phase will be helpful to find malicious node in forwarding group.as here destination continuously monitor two parameter that are end to end delay and jitter parameter and match their result with threshold decided for that network.

If destination detect high variation around threshold than it detects that there would be something went wrong in the forwarding path and this mechanism detects jellyfish delay variance attack in second phase

Here we have a two threshold value for each phase firstly we have first phase threshold which is depends upon processing delay of the node which we could set 1 -1000 micro second depends upon the different parameter of network .than for second phase we have a value set according to maximum delay produced due to different types of delay available in the network

$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

The node processing delay d_{proc} is the time that a node spends processing a packet. This includes time for error checking, time for reading the packet header, and time for looking up the link to the next node , based on the destination address. Although the processing may sound complicated, the nodal processing delay is usually negligible compare to another other item in the delay equation

The transmission delay d_{trans} is the time required to put an entire packet into the communication media. It can be computed by the following equation.

$d_{trans} = L/R$

here L is length of a packet in bits and R is the transmission rate in bits per time unit .the time unit in dtrans should be the same

The queuing delay dque is the time that a packet spend in a queue at a node while waiting for other packet to be transmitted .If the node is a high speed router then there is one queue for each outgoing link, so a packet waits only for other packets that are going across the same link.

dqueue = dtrans * l queue

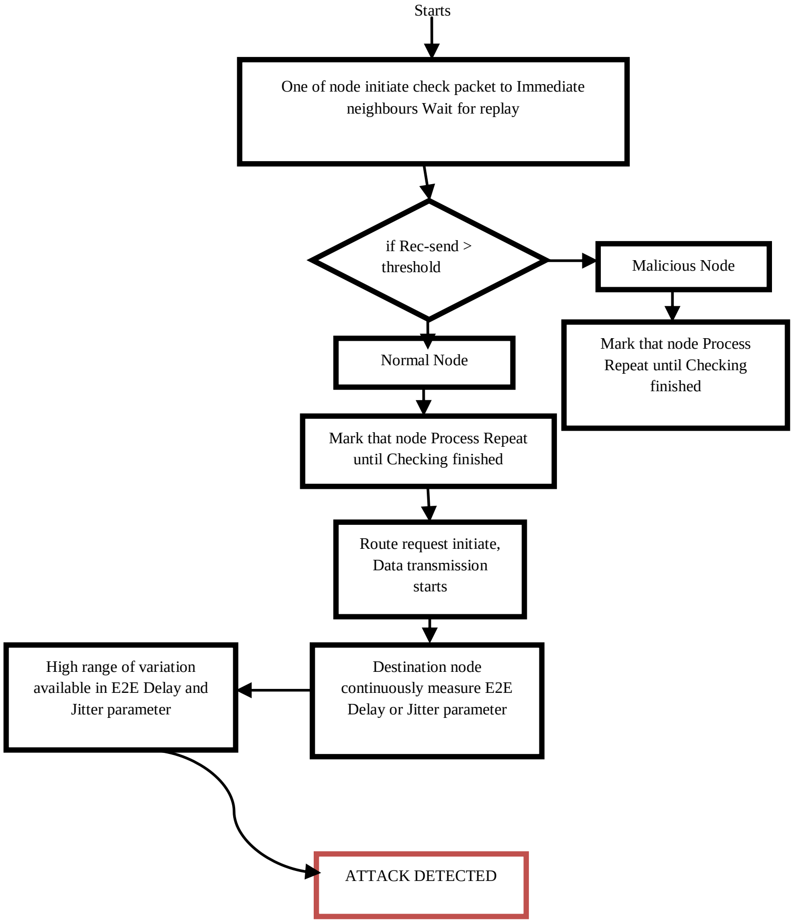
The queuing delay is related to the transmission rate .The average queue length is typically less than 1 for a load factor less than ½ .when the load factor exceeds 1 ,the queue length grows without bound.

The propagation delay is the time that it takes a signal change to propagate through the communication media from a node to the next node .it can be computed using the following equation

$d_{prop} = D/s$

Here D is the distance from the node to the net node and s is propagation speed of the media for links using radio broadcast , a signal changes propagates at close to the speed of light ,which is about 186,000 mile per second. for copper and fiber links , a signal changes propagate at 60% to the 80% of the speed of light

*A. **Diagrammatic representation of the proposed detection method***


Starts
One of node initiate check packet to Immediate neighbours Wait for replay
if Rec-send > threshold
Malicious Node
Mark that node Process Repeat until Checking finished
Normal Node
Mark that node Process Repeat until Checking finished
Route request initiate, Data transmission starts
Destination node continuously measure E2E Delay or Jitter parameter
High range of variation available in E2E Delay and Jitter parameter
ATTACK DETECTED

B. PROPOSED PREVENTION METHOD

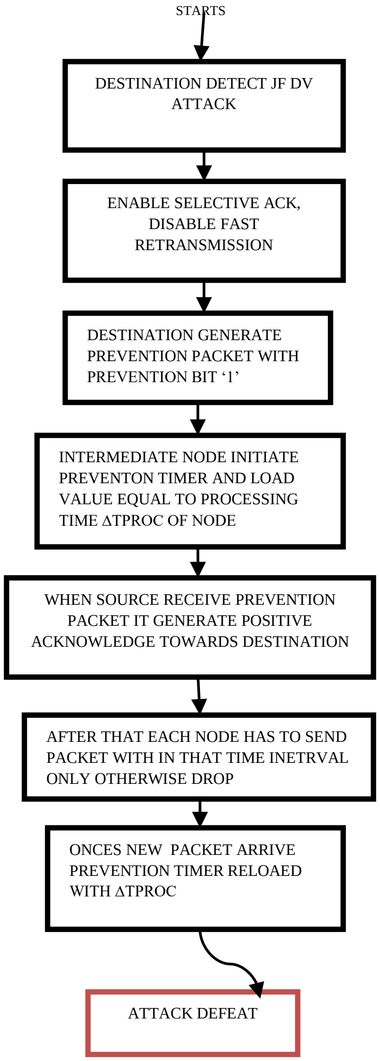
In this method when destination detects this variation It firstly start selective acknowledge and stop fast retransmission .then destination generate prevention packet in which it sets prevention bit “1” from “0” and sends this packet to source towards existing path And when the intermediate node receive this packet and detect its prevention bit “1” than it start loading Prevention timer to value equal to the time require to process a regular whose value is specified in prevention packet .so when this packet is received by intermediate nodes than they automatically load their timer by that value .

This way each node within the forwarding path doing the same process and including source ,Now concept is that each node has to send packet in that time interval only ,intermediate node has to send that packet in that time interval only ,so if attacker produce unwanted delay that could not effect to end to end delay.so we can achieve our average end to end delay back even-though attacker node is present in the forwarding path.

More precisely suppose our threshold for phase two is 8.5 msand so any high variation around that value is detected by destination then destination generate prevention packet with prevention bit set to “1” and value of processing a packet that destination take to process packet suppose destination take 1ms to process each data packets so its adds this 1ms value as a timer reload value and start to send that packet to immediate neighbor of existing forwarding path .when neighbor receive this packet and detect prevention bit value “1” then it load prevention timer that it has to a value of 1ms .and its takes this value from prevention packet which has sent by destination Same process repeated by each intermediate node including source. when source reload its counter than its generate a positive acknowledgement for destination that it has receive prevention packet and load a counter successfully , so from that onwards when each node receive a packet of forwarding group then timer start decrementing and when its become zero than that node sends packet to its next node of forwarding path without any dealy.so in our case each node takes 1 ms as we take a example of 1 ms of reload value

The benefits of such method is that even-though attacker available in the forwarding path and made random delay but it would not effects because each node sends packets within the time interval of the prevention timer. This way MANET could not disturb by the jellyfish delay variance attack.

B. *Diagrammatic representation of the proposed prevention method*


STARTS
DESTINATION DETECT JF DV ATTACK
ENABLE SELECTIVE ACK, DISABLE FAST RETRANSMISSION
DESTINATION GENERATE PREVENTION PACKET WITH PREVENTION BIT ‘1’
INTERMEDIATE NODE INITIATE PREVENTON TIMER AND LOAD VALUE EQUAL TO PROCESSING TIME ΔTPROC OF NODE
WHEN SOURCE RECEIVE PREVENTION PACKET IT GENERATE POSITIVE ACKNOWLEDGE TOWARDS DESTINATION
AFTER THAT EACH NODE HAS TO SEND PACKET WITH IN THAT TIME INETRVAL ONLY OTHERWISE DROP
ONCES NEW PACKET ARRIVE PREVENTION TIMER RELOAED WITH ΔTPROC
ATTACK DEFEAT

C. METHODOLOGY AND SIMULATION DESIGN

For the proposed work we are using the following methodology:

Here we are tacking three scenario for different node density E.g 25 nodes,50 nodes, 75 nodes

Following table shows the analysis for 25 nodes network.

Malicious Node Density	End to End Delay(ms)
0	8.54
1	13.91
2	16.48
3	23.74
4	84.30
5	30.43
6	95.05

Above table shows relationship between numbers of malicious node density and its associate end to end delay this results shows that as numbers of malicious nodes increase amount of end to end delay increase which is our detection factor

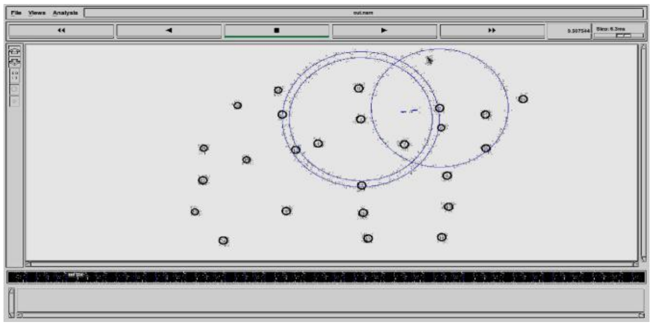


Fig .1 MANET in Normal case

For experimental purpose we have simulated a Mobile Ad Hoc Network under delay variance JF attack using NS2 simulator. We are using the above simulation scenarios in this paper:

In figure 1 we use 25 mobile nodes and build a scenario without any JF attacker. It’s a normal flow of traffic.

In figure 2 we use 25 mobile nodes and build a scenario with four JF attackers. JF attackers are shown in red label i.e. attacker1, attacker2 etc.

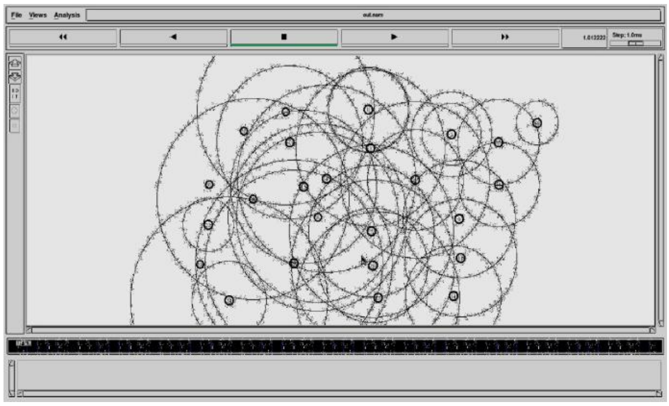


Fig 2 MANET under JF delay variance attack

D. *Experiment Design Parameters*

1> common parameter

TABLE I
COMMON PARAMETERS USED IN SIMULATION

Parameter	Value
Platform	Windows 10
Simulator	Ns2
Area	1*1 KM (FIX)
Node size	25 node(FIX)
Mobility model	Random
Traffic type	FTP
Simulation time	1 minute
Address mode	IPV4
Ad-hoc Routing Protocol	AODV
AODV Parameters	Default
Jellyfish Attacker	Zero for Normal flow Five for attacking flow

E.MANET Traffic Generation Parameters:

TABLE II

2> MANET TRAFFIC GENERATION PARAMETERS USED IN SIMULATION

Parameters	Value
Start time(ms)	0.1ms
Packet arrival time	Exponential
Packet size	Constant(1024)
Destination IP add	Random
Stop time	End of Simulation

3) Wireless LAN Parameters:

TABLE III
WIRELESS LAN PARAMETERS USED IN SIMULATION

Parameter	Value
BSS Identifier	Auto Assigned
Physical characteristic	Direct Sequence
Data rate	0.1 Mbps
Channel setting	Auto Assigned
Transmit power	0.005
Threshold	Phase 1 - 1ms
	Phase 2- 8.5ms
Buffer size	256000
Other parameter	Default

E. Results

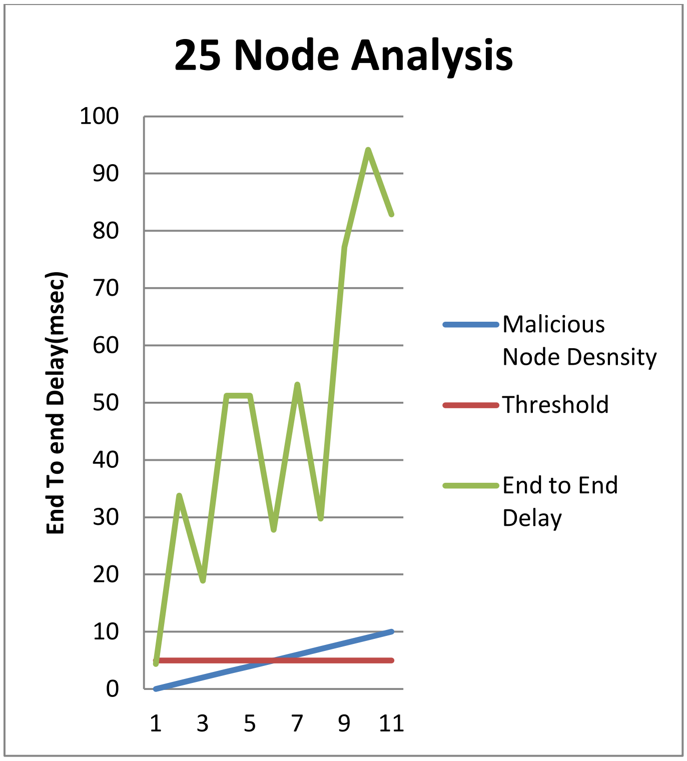


Fig 3 End to End Delay

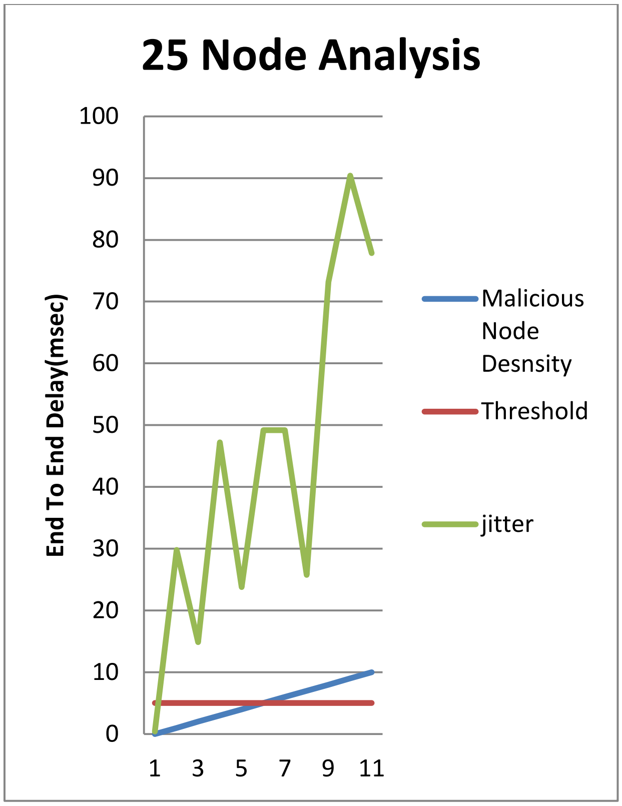


Fig 4 JITTER

V. CONCLUSION

Use of Transport protocol e.g TCP is biggest challenge in this type network. Because it makes network more vulner to this attack. AODV is more effective compare to DSR in detection and prevention for jellyfish delay variance attack when specific delay is being concern. Data traffic and mobility , amount of intrusion of the nodes plays biggest role to this attack to be affected.

APPLICATION

One of the applications of our work is in war field. Suppose war is going on there is the probability that we have to increase the number of nodes in the network. Enemy wants to do JF attack for producing long delay in information exchange so that information can't reach to destination within time limit.

ACKNOWLEDGE

We hereby certify that we are the authors of this research and that neither any part of this research has been submitted for any other research entities. we highly obliged our honourable guide have devoted his valuable time and shared his expertise knowledge we extend our sincere thanks to faculties of Department of Electronics and Communication Engineering and Dean, **IITE, Indus University**, Ahmedabad as well as **Silver-Owk Institute of Technology** for providing us such an opportunity to do my research work.

REFERENCES

[1] Jan vonMulert, IanWelch, WinstonK.G.Seah, “Security threats and solutions in MANETs: A case study using AODV and SAODV”, *Elsevier Journal of Network and Computer Applications* 35 (2012) 1249–1259.

[2] Syed Atiya Begum, L.Mohan, B.Ranjitha, “ Techniques for Resilience of Denial of Service Attacks in Mobile Ad Hoc Networks”, *Proceedings published by International Journal of Electronics Communication and Computer Engineering* Volume 3, Issue (1) NCRTCST, ISSN 2249 –071X *National Conference on Research Trends in Computer Science and Technology* – 2012.

[3] Ekta Barkhodia, Parulpreet Singh, Gurleen Kaur Walia, “Performance Analysis of AODV using HTTP traffic under Black Hole Attack in MANET”, *Computer Science & Engineering: An International Journal (CSEIJ),* Vol.2, No.3, June 2012.

[4] Uttam Ghosh, Raja Datta, “Identity based Secure AODV and TCP for Mobile Ad Hoc Networks”, *Proceedings of ACM ACWR’11,* December 18 - 21 2011.

[5] Ahmed Khurshid, Md. Humayun Kabir, Md. Anindya Tahsin Prodhan, “An Improved TCP Congestion Control Algorithm for Wireless Networks”, *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PacRim),* 2011.

[6] B. B. Jayasingh, B. Swathi, “A Novel Metric For Detection of Jellyfish Reorder Attack on Ad Hoc Network*”, BVICAM’S International Journal of*

Information Technology (BIJIT) Vol. 2 No. 1, ISSN 0973 – 5658 Year – 2010.

[7] Vishnu K, “A new kind of Transport Layer attack in Wireless Ad Hoc Networks”, *IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS)*, 2010.

[8] Hosam El-Ocla, “TCP CERL: congestion control enhancement over wireless networks”, *ACM Journal on Wireless Networks* Vol 16. Jan 2010.

[9] Hoang Lan Nguyen, Uyen Trang Nguyen “A study of different types of attacks on multicast in mobile ad hoc networks”, *Elsevier Journal of Ad Hoc Networks* 6 (2008) 32–46.

[10] Imad Aad, JeanPierre Hubaux, Edward W. Knightly, “Denial of Service Resilience in Ad Hoc Networks”, *In Proceedings of ACM MobiCom’04,* Sept. 26 Oct.1, 2004.

[11] S. Bohacek, J. Hespanha, J. Lee, C. Lim, K. Obraczka, “TCP-PR: TCP for persistent packet reordering”, *In Proceedings of the 23rd IEEE International Conference on Distributed Computing Systems*, May 2003.

[12] Wazid, Mohammad, Avita Katal, Roshan Singh Sachan, and R. H. Goudar. "*E-TCP for Efficient Performance of MANET under JF Delay Variance* ***Attack.***" 2013 IEEE Conference On Information And Communication Technologies (2013)