Intrusion Detection-Watchdog: For Secure AODV Routing Protocol in VANET

Nishant P Makwana[1] **, Sunil K Vithalani**[2] **, Jayesh D Dhanesha**[3]

[1](Department of Computer Engineering (Wireless & Mobile Computing), **GTU-**Gujarat Technological University, Ahmedabad-**360 015**

[2](Department of Computer Engineering, **GTU-**Gujarat Technological University, Ahmedabad-**360 015**

[3](Department of Computer Engineering, **GTU-**Gujarat Technological University, Ahmedabad-**360 015**

ABSTRACT

Vehicular Ad hoc Network (VANET) needs security to implement the wireless environment and serves users with safety and comfort applications. Attackers generate different attacks in vehicular network. In this paper, first phase implementation of attacker node in AODV routing protocol in VANET and in second phase identify malicious node with watchdog intrusion detection system. Once the attacker node is identified we will prevent it to communicate with other neighbor nodes in network with the help of Beyesian network theory. From Beyesian network theory find probability of the neighbouring node being attacker node. To make secure AODV connection with generate new Route Request Packet (RREQ) in VANET.

Keywords **- VANET, AODV, SUMO, NS2, Watchdog, RREQ, Beyesian Network**

I. INTRODUCTION

VANET (Vehicular Ad Hoc Networks) is a recent advances in wireless networks have led to the introduction of a new type of networks. VANETs [1] is subclass of Mobile Ad Hoc Networks (MANETs). VANET nodes having high mobility than MANETs network. VANETs provide us with the infrastructure for developing new systems to enhance drivers' and passengers' safety and comfort. VANETs are distributed self organizing networks formed between moving vehicles equipped with wireless communication devices. VANETs possess a few distinguishing characteristics from MANETs. These are:

- Patterned Mobility
- Highly dynamic topology
- Propagation Model
- Unlimited Battery Power and Storage.
- On-board Sensors.

There are many routing protocols that have been proposed and assessed to improve the efficiency of VANET.

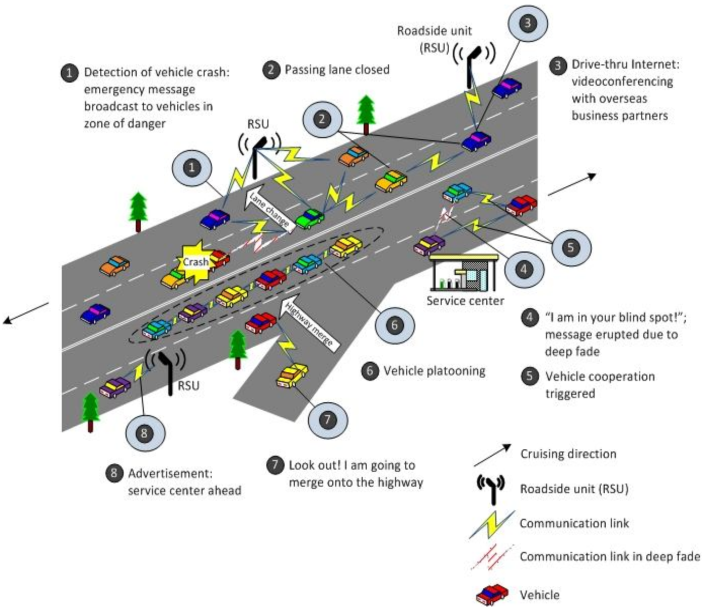

Roadside unit (RSU)
3
3 Drive-thru Internet: videoconferencing with overseas business partners
1 Detection of vehicle crash: emergency message broadcast to vehicles in zone of danger
2 Passing lane closed
RSU
2
1
Lane change
Crash
Highway merge
Service center
5
4
6
RSU
7
8
4 "I am in your blind spot!"; message erupted due to deep fade
5 Vehicle cooperation triggered
6 Vehicle platooning
7 Look out! I am going to merge onto the highway
8 Advertisement: service center ahead
Cruising direction
Roadside unit (RSU)
Communication link
Communication link in deep fade
Vehicle

Fig 1 VANET Scenario

II. AODV

- AODV, source node and intermediate nodes store the next hop information corresponding to each flow for data packet transmission.

- on-demand routing protocol, the source code floods the RouteRequest packet in the network when a route is not available for the desired destination.
- It may obtain multiple routes to different destinations from a single RouteRequest.
- AODV uses a destination sequence number DestSeqNum to determine an uptodate path to the destination.
- A node updates its path information only if the DeptSeqNum of the current packet received is greater than the last DeptSeqNum stored at the node.
- A RouteRequest carries source identifier (SrcID), the destination identifier (DestID), the source sequence number (SrcSeqNum),destination sequence number (DestSeqNum), the broadcast identifier (BcastID), time to live (TTL) field.
- DestSeqNum indicates the freshness of the route that is accepted by the source.
- Validity of a route at the intermediate node is determined by comparing the sequence number at the intermediate node with the destination sequence number in the RouteRequest packet.
- If RouteRequest is received multiple times, indicated by BcastID-SrcID pair, the duplicate copies are discarded.
- A timer is used to delete this entry in case RouteReply is not received before the timer expires.

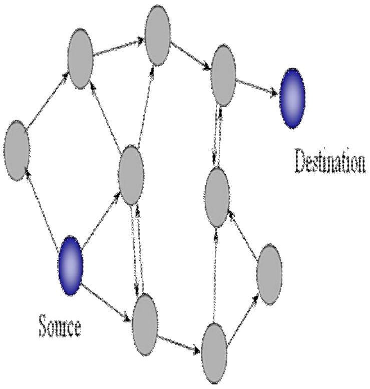


Fig 2(a) Route Request (RREQ)

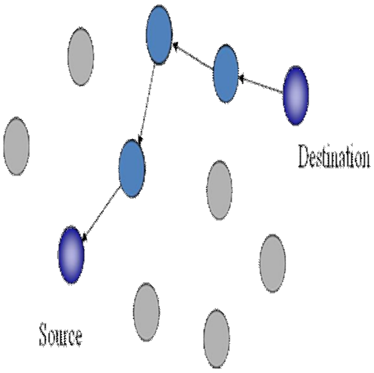


Fig 2(b) Route Reply (RREP)

III. RESEARCH METHODOLOGY USED

A. Simulation tools

The simulation module created using TCL makes use of two tools to simulate the implementation and evaluate its performance:

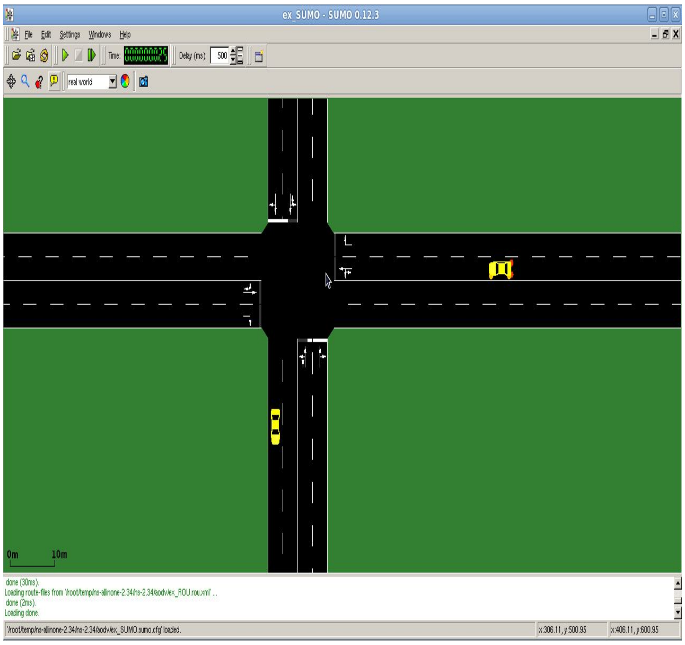
1) MOVE: MObility model generator for Vehicular networks [5], [6] tool is used to facilitate users to rapidly generate realistic mobility models for VANET simulations. MOVE is currently implemented in java and is built on top of an open source micro-traffic simulator SUMO. By providing a set of Graphical User Interfaces that automate the simulation script generation, MOVE allows the user to quickly generate realistic simulation scenarios without the hassle of writing simulation scripts as well as learning about the internal details of the simulator.

The output of MOVE is a mobility trace file that contains information about realistic vehicle movements which can be immediately used by popular simulation tools such as ns-2.

2) NS2: The Network Simulator (ns2) [7] is a discrete event driven simulator developed at UC Berkeley. We are using Network Simulator NS2 for simulations of protocols. It provides substantial support for simulation of TCP, routing and multicast protocols over wired and wireless networks. Ns-2 code is written either in C++ and OTCL and is kept in a separate file that is executed by OTCL interpreter, thus generating an output file for NAM (Network animator) . It then plots the nodes in a position defined by the code script and exhibits the output of the nodes communicating with each other.

It consists of two simulation tools. The network simulator (ns) contains all commonly used IP protocols. The network animator (NAM) is use to visualize the simulations.

3) SUMO: "Simulation of Urban MObility" (SUMO) [9] is an open source, highly portable, microscopic road traffic simulation package designed to handle large road networks. It allows the user to build a customized road topology, in addition to the import of different readymade map formats of many cities and towns of the world. Fig.-3 shows SUMO visualization.



```
done (30ms).
Loading route-files from '/root/temp/ns-allinone-2.34/ns-2.34/aodv/ex_ROU.rou.xml' ...
done (2ms).
Loading done.
'/root/temp/ns-allinone-2.34/ns-2.34/aodv/ex_SUMO.sumo.cfg' loaded.    x:306.11, y:500.95    x:406.11, y:600.95
```

Fig. 3 SUMO Visualization

B. Simulation configuration

Parameter	Value
Channel Type	ChannelWirelessChannel
Network Interface Type	PhyWirelessPhy
Routing Protocol	AODV
Interface Queue Type	QueueDropTail/PriQueue
No of Node in Topology	24
No of Flow in Topology	2 flow0_0 to 0_11 & flow1_0 to 1_11
No of malicious Node in Topology	4 [Ref-14]
X and Y Dimension of Topology	652 * 752
Time of Simulation end	999.00
Traffic Type	TCP
MAC Type	IEEE802.11
Radio Propagation Model	Propagation/Two Ray Ground
NAM trace file	ex_NS2.nam
Trace o/p file	ex_NS2.tr

Table-1 SUMO Setup

IV. ADD MALICIOUS NODE

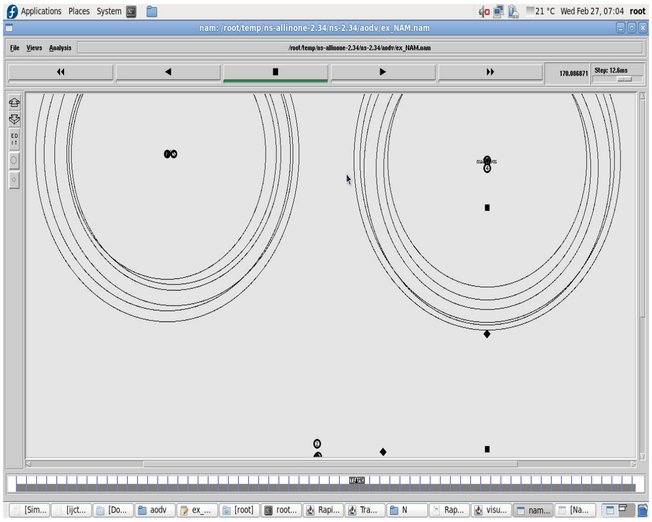


Fig 4 Add malicious node

Fig 4 capture after adding malicious node in VANET scenario which is simply drops the packets from the neighbor node. In **V** Part measure QoS parmeter before adding malicious node Vs after adding malicious node.

V. SIMULATION PARAMETER

1) PDR: Packet Delivery Ratio = Total Packets Received / Total Packets Sent.

The ratio of the number of data packets successfully delivered to the destinations to those generated by CBR sources. Packet delivery ratio describes the loss rate.

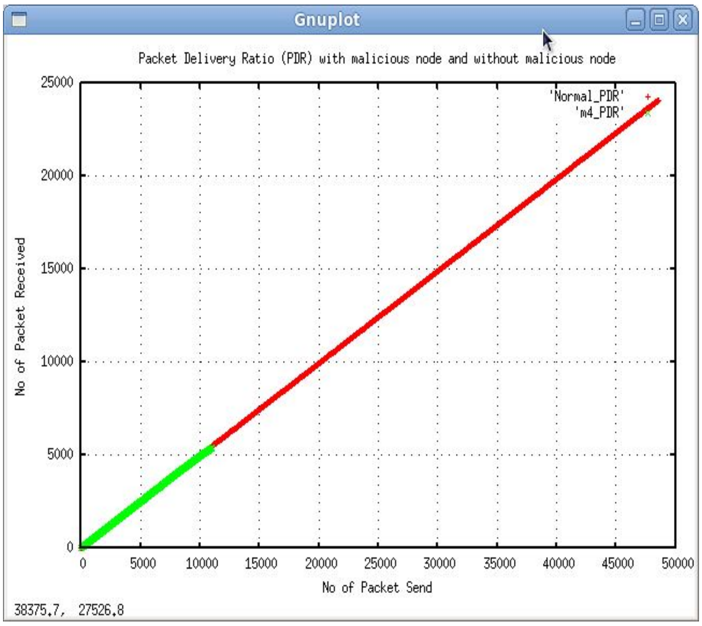


Fig 5(a) Packet Delivery Ratio

As Show in Fig 5(a) PDR ratio of Normal_PDR (without malicious node PDR) and m4_PDR (malicious PDR) with malicious node and without malicious node.

Normal_PDR (without malicious node) PDR is:

GeneratedPacket = 48590
ReceivedPacket = 24103
Packet Delivery Ratio = 49.604857

m4_PDR (with malicious node) PDR is:

GeneratedPacket = 10966
ReceivedPacket = 5382
Packet Delivery Ratio = 49.078971

2) End - to - End Delay:

"Average Time taken by data packet to arrive in the destination"

End-to-end Delay = Arrive Time – Send Time / No of Connection

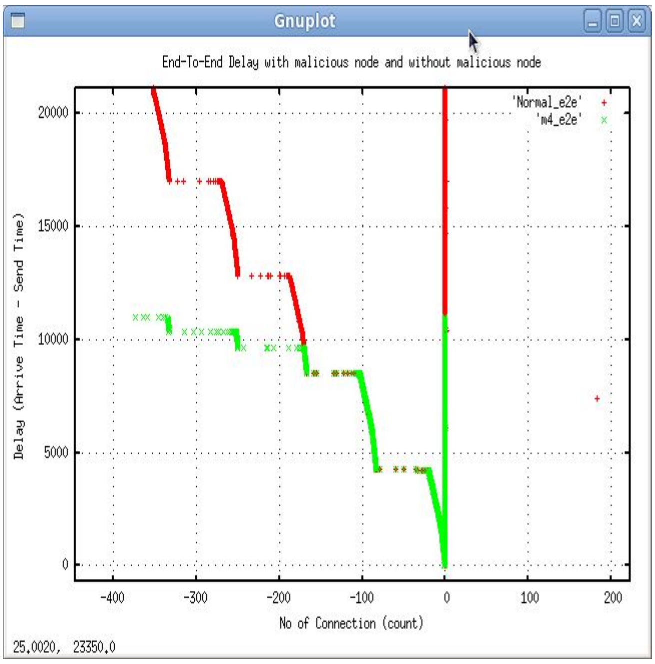


Fig 5(b) End-to-end Delay

As Show in Fig 5(b) end-to- end Delay of Normal_PDR (without malicious node) and m4_PDR (with malicious node) with malicious node and without malicious node.

Normal_e2e (without malicious node) E2E is:

GeneratedPacket	= 48590
ReceivedPacket	= 24103
Total Dropped VANET Packet	= 427
Average End-to-end Delay	= - 224000ms

m4_e2e (with malicious node) E2E is:

GeneratedPacket	= 10966
ReceivedPacket	= 5382
Total Dropped VANET Packet	= 228
Average End-to-end Delay	= - 46835ms

VI. INTRUSION DETECTION SYSTEM (IDS)

An intrusion detection system is a software/hardware tool used to detect unauthorized accesses to a computer system or a network. [17]

A. Mechanism - WathDog

The watchdog identifies misbehaving nodes, while the bathwater avoids routing packets through these nodes. When a node forwards a packet, the node's watchdog verifies that the next node in the path also forwards the packet [16].The watchdog does this by listening promiscuously to the next node's transmissions. If the next node does not forward the packet, then it is misbehaving.

B. WatchDog

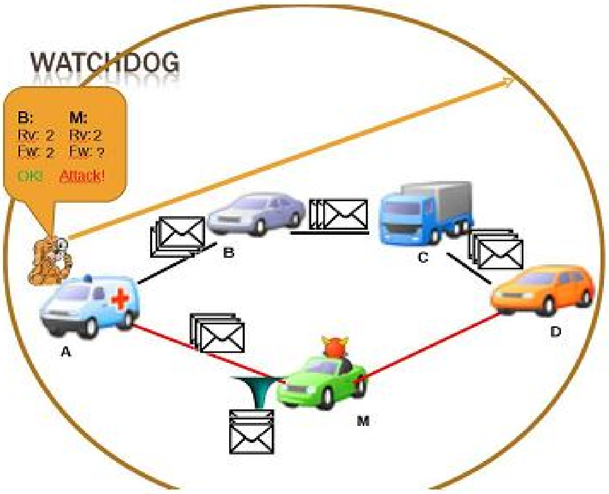


Fig 6.1 WatchDog

The watchdog method detects misbehaving nodes. Figure6.1 (a) illustrates how the watchdog works. Node A cannot transmit all the way to node C, but it can listen in on node B's traffic. Thus, when A transmits a packet for B to forward to C.

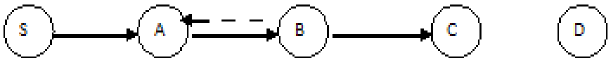


Fig 6.1(a) How WatchDog works

A can often tell if B transmits the packet. If encryption is not performed separately for each link, which can be expensive, then A can also tell if B has tampered with the payload or the header.

We implement the watchdog by maintaining a buffer of recently sent packets and comparing each overheard packet with the packet in the buffer to see if there is a match. If so, the packet in the buffer is removed and forgotten by the watchdog, since it has been forwarded on. If the packet has remained in the buffer for longer than a certain timeout, the watchdog increments a failure tally for the node responsible for forwarding on the packet. If the tally exceeds a certain threshold bandwidth, it determines that the node is misbehaving and sends a message to the source notifying it of the misbehaving node.

Advantages

The watchdog mechanism can detect misbehaving nodes at forwarding level and not just the link level.

Weakness

It might not detect misbehaving nodes in presence of 1) ambiguous collusions 2) receiver collusions 3) limited transmission power 4) false misbehavior 5) collision 6) partial dropping.

C. Analysis of Watchdog's weaknesses

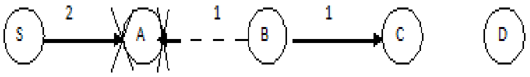


Fig 6.1(c) Ambiguous Collision

1) Ambiguous collision

The ambiguous collision [18] problem prevents A from overhearing transmissions from B. As figure 6.1(c) illustrates, a packet collision occur at A while it is listening for B to forward on a packet. A does not know if the collision was caused by forwarding on a packet as it should or if B never forwarded the packet and the collision was caused by other nodes in A's neighborhood. Because of this uncertainty, A should instead continue to watch B over a period of time.

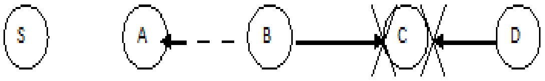


Fig 6.1 (d) Receiver Collision

2) Receiver collision

In the receiver collision [18] problem, node A can only tell whether B sends the packet to C, but it cannot tell if C receives it. If a collision occurs at C when B first forwards the packet, A only sees B forwarding the packet and assumes that C successfully receives it. Thus, B could skip retransmitting the packet and evade detection. Figure 6.1(d)

3) False misbehavior

False misbehavior can occur when nodes falsely report other nodes as misbehaving. A malicious node could attempt to partition the network by claiming that some nodes following it in the pat h are misbehaving. For instance, node A could report that node B is not forwarding packets when in fact it is. This will cause S to mark B as misbehaving when A is the culprit. This behavior, however, will be detected. Since A is passing messages onto B (as verified by S), then any acknowledgements from D to S will go through A to S, and S will wonder why it receives replies from D when supposedly B dropped packets in the forward direction. In addition, if A drops acknowledgements to hide them from S, the node B will detect this misbehavior and will report it to D.

4) Limited transmission power

Another problem is that a misbehaving node that can control its transmission power can circumvent the watchdog. A node could limit its transmission power such that the signal is strong enough to be overheard by the previous node but too weak to be received by the true recipient.

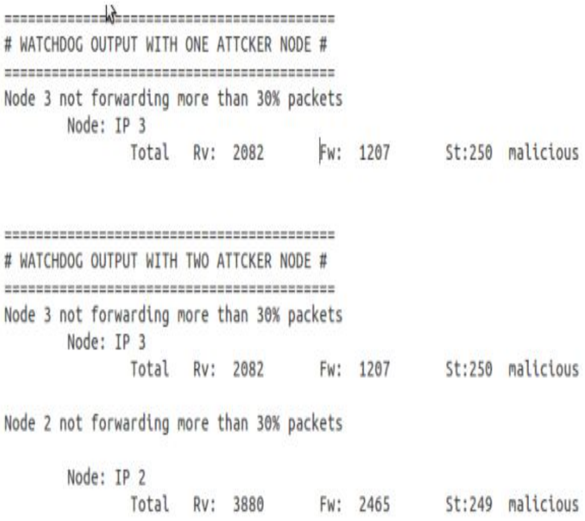
5) Multiple colluding nodes

Multiple nodes in collusion can mount a more sophisticated attack. For example, B and C could collude to cause mischief. In this case, B forwards a packet to C but does not report to A when C drops the packet. Because of its limitation, it may be necessary to disallow two consecutive untrusted nodes in a routing path.

6) Partial dropping

A node can circumvent the watchdog by dropping packets at a lower rate than the watchdog’s configured minimum misbehavior threshold. Although the watchdog will not detect this node as misbehaving, this node is forced to forward at the threshold bandwidth. In this way the watchdog serves to enforce this minimum bandwidth. For the watchdog to work properly it must know where a packet should be in two hops.

D. Detection of Malicious Node



```
==========================================
# WATCHDOG OUTPUT WITH ONE ATTCKER NODE #
==========================================
Node 3 not forwarding more than 30% packets
        Node: IP 3
                Total   Rv:  2082       Fw:  1207       St:250  malicious


==========================================
# WATCHDOG OUTPUT WITH TWO ATTCKER NODE #
==========================================
Node 3 not forwarding more than 30% packets
        Node: IP 3
                Total   Rv:  2082       Fw:  1207       St:250  malicious

Node 2 not forwarding more than 30% packets

        Node: IP 2
                Total   Rv:  3880       Fw:  2465       St:249  malicious
```

Fig 7 WatchDog Output

E. BEYESIAN NETWORK THEORY

Bayesian networks are known to be used for calculating new beliefs when new information (evidence) is available. The basic task of the inference system is to compute the Posterior probability upon arrival of some evidences. This is called belief updating or probabilistic inference.[18]

VII. Conclusion & Further Work

I analyze the performance of AODV with and without malicious node under the circumstances of different parameters. Simulation results show, that when a node becomeas a malicious node it will effect on the AODV performance. The route discovery process in the AODV is susceptible to malicious node and therefore, it is vital to have an efficient security functions in the protocol in order to avoid such attacks.

First perform the solution for the malicious node and other attacks like blackhole, grayhole or other and apply this for AODV and measure different QoS(Quality of Service) parameter.

For detect unauthorized accesses to a computer system or a network **Watchdog** is implemented in AODV with blackhole attack.

In further work, from **Beyesian Network Theory** find probability of the neighbouring node being attacker node for to make Secure AODV connection in VANETs.

REFERENCES

[1] Brijesh Kumar Chaurasia, Shekhar Verma "Attacks on Anonymity in VANET"- *209 International Conference on Computational Intelligence and Communication Systems*

[2] Irshad Ahmed Sumra,Iftikhar Ahmad, Halabi Hasbullah,"*Classes of Attacks in VANET*"

[3] Sandhaya Kohli, Bandanjot Kaur, Sabina Bindra "A comparative study of Routing Protocols in VANET"

[4] F. Maan, Y. Abbas, N. Mazhar "*Vulnerability Assessment of AODV and SAODV Routing Protocols Against Network Routing Attacks and Performance Comparisons*"

[5] Tajinder Kaur, A. K. Verma "*Simulation and Analysis of AODV routing protocol in VANETs" - Volume-2, Issue-3, July 2012*

[6] Harris Simaremare, Riri Fitri Sari "*Performance Evaluation of AODV variants on DDOS, Blackhole and Malicious Attacks*" *-VOL.11 No.6, June 2011*

[7] Sushil Kumar Chamoli, Santosh Kumar "*Performance of AODV against Black Hole Attacks in Mobile ad-hoc Networks " - Vol 3 (4), 1395-1399*

[8] Ghassan Samara, Wafaa A.H. Al -Salihy, R. Sures National Advanced IPv6 Center, Universities Sains Malaysia Penang, Malaysia "*Security Analysis of Vehicular Ad Hoc Networks*" *- 2010 Second International Conference on Network Applications, Protocols and Services*

[9] Vijay Kumar, Rakesh Kumar "*Effect of Malicious Nodes on AODV in Mobile Ad Hoc Networks*" *- Vol 1 Issue 3 October 2012*

[10] SUMO User Documentation**:** http://sumo.sourceforge.net/

[11] Rapid Generation of Realistic Simulation for VANET Manual http://lens1.csie.ncku.edu.tw/MOVE/index.htm

[12] A Tutorial on the Implementation of Ad-hoc OnDemand Distance Vector (AODV) Protocol in Network Simulator (NS-2)

[13] AWK : http://mohittahiliani.blogspot.in/2009/12/awk-script-for-ns2.html

[14] Add malicious node in AODV: http://narentada.com/what-is-black-hole-attack-in-manets-my-code-for-adding-malicious-node-as-blackhole-in-aodv-protocol/

[15] GNU Plot : https://sites.google.com/a/seecs.edu.pk/network-technologies-tcp-ip-suite/home/analysis-of-performance-of-broadcast-traffic-in-multi-radio-multi-channel-multi-rate-wireless-mesh-networks-using-ns-3-sarfraz-ahmed/implementation/handling-output-trace-files

[16] Sergio Marti, T.J. Giuli, Kevin Lai, and Mary Baker "*Mitigating Routing Misbehavior in Mobile Ad Hoc Networks*"

[17] Jorge Hortelano, Juan Carlos Ruiz, Pietro Manzoni "*Evaluating the uselfusness of watchdogs for intrusion detection in VANETs*"

[18] Mahmoud Abuelela Stephan Olariu "*Automatic Incident Detection In VANETs: A Bayesian Approach*"

[19] Sanjay Sharma Abhinav Jain and Mahendra Singh Sisodia "*Network Intrusion Detection By using Supervised and Unsupervised Machine Learning Techniques: A Survey*". *In:International Journal of Computer Technology and Electronics Engineering. 2011.*

[20] Christian CALLEGARI. "*Network Security: Attacks and Defenses*".

[21] Yu-Feng Hsu Chih-Fong Tsai. "*Intrusion detection by machine learning: A review". In: Expert Systems with Applications 36. Expert Systems with Applications 36. Elsevier, 2009, pp. 11994{12000}*

[22] S. Zair M. Mehdi, A. Anou, and M. Bensebti. "*A Bayesian Networks in Intrusion Detection Systems". In: Journal of Computer Science 3 (5): 259-265. Science Publications, 2007*