**INTELLIGENCE INTRUSION DETECTION USING GLOWWORM SWARM OPTIMIZATION WITH MULTINOMIAL NAÏVE BAYESIAN TECHNIQUE IN MANET**

**PROJECT REPORT**

***Submitted by***

**B.JEYA SURYA(810015104030)**

**R. MONISHA(810015104047)**

***Under the guidance of***

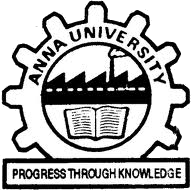
**Mr.M.KRISHNA KUMAR**

***in partial fulfilment for the award of the degree*  *of***

**BACHELOR OF ENGINEERING**

***In***

**COMPUTER SCIENCE AND ENGINEERING**

****

**UNIVERSITY COLLEGE OF ENGINEERING –BIT CAMPUS**

**TIRUCHIRAPALLI**

**ANNA UNIVERSITY, CHENNAI -600 025**

**APRIL 2019**

**i**

**DECLARATION**

We hereby declare that the work entitled **“INTELLIGENCE INTRUSION DETECTION USING GLOWWORM SWARM OPTIMIZATION WITH MULTINOMIAL NAÏVE BAYESIAN TECHNIQUE IN MANET”** is submitted in partial fulfilment of the requirement for the award of the degree in B.E-Computer Science and Engineering, Bharathidasan Institute of Technology, Anna University, Tiruchirappalli, is a record of the our original work carried out by us during the academic year 2014-2015. Under the supervision and guidance of Mr.M.KRISHNA KUMAR, Teaching Fellow, Bharathidasan Institute of Technology, Anna University, Tiruchirappalli. The extent and source of information are derived from the existing literature and have been indicated through the dissertation at the appropriate places. The matter embodied in this work is original and has not been submitted for the award of any other degree or diploma, either in this or any other university. B.JEYA SURYA

810015104030

R.MONISHA

810015104047

I certify that the declaration made above by the candidate is true.

Mr. M.KRISHNA KUMAR

Teaching Fellow

Department of Computer Science & Engineering

Bharathidasan Institute of Technology

Anna University

Tiruchirappalli

**ii**

**BONAFIDE CERTIFICATE**

Certified that this project report titled “**INTELLIGENCE INTRUSION DETECTION USING GLOWWORM SWARM OPTIMIZATION WITH MULTINOMIAL NAÏVE BAYESIAN TECHNIQUE IN MANET***”,* is a bonafide work of **Ms. B.JEYA SURYA** and **Ms. R.MONISHA** work under my supervision, for the partial fulfilment of the requirements for the award of the degree of *Bachelor of Engineering in Computer Science and Engineering*. Certified further that to the best of my knowledge and belief, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion.

**SIGNATURE SIGNATURE**

Mr. D. VENKATESAN Mr.M.KRISHNA KUMAR

**HEAD OF THE DEPARTMENT SUPERVISOR**

Professor Teaching Fellow

Department of Computer Science & Department of Computer

Engineering Science & Engineering

Bharathidasan Institute of Bharathidasan Institute of

Technology Technology

Anna University Anna University

Tiruchirappalli-620 024. Tiruchirappalli-620 024.

Submitted for the B.E degree examination held on

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**iii**

**ACKNOWLEDGEMENT**

First and foremost we would like to thank the God for giving us the power to believe in our self and pursue our dreams.

We wish to express our profound thanks with gratitude to **Mr.M.KRISHNA KUMAR**, Teaching Fellow, Department of Computer Science & Engineering, for providing us the opportunity to do this project.

We wish to express our gratitude to the officials and other staff members of our department who rendered their help during the period of our project work.

Last but not least we wish to avail ourselves of this opportunity, express a sense of gratitude and love to our friends and our beloved parents for their manual support, strength, and help for everything.

JEYA SURYA.B

810015104030

R.MONISHA

810015104047

**iv**

**ABSTRACT**

A Mobile Adhoc Network is a self-configuring, self-healing, infrastructureless network whenever every node act as router for different nodes which permits knowledge to travel, utilizing multihop network paths. Due to unique characteristics of Mobile Adhoc Network, these are more vulnerable to severe attacks such as Blackhole attack, Selective packet dropping attack, flooding attack .An Anomaly intrusion detection which is the identification of observations that do not conform to the expected pattern of a given group. The Intelligent Intrusion Detection System using Multinomial Naive Bayesian concept with Glowwarm Swarm Optimization. Multinomial Naive Bayesian algorithm used to classify the trust nodes in Network and Glowworm Swarm Optimization is used for route optimization based on the fitness value of the node. Proposed system increases the performance of a network in terms of numerous parameters like Throughput, Overhead, Delay and Packet Delivery Ratio which is also decrease the false positive rate of Adhoc Networks.

**v**

**LIST OF FIGURES**

**FIGURE NO NAME OF FIGURE PAGE**

1.2.1 ATTACKS IN MANET 2

1.3.1 TYPES OF IDS 2

3.3.1 PROPOSED ARCHITECTURE 10

4.3.1 NS2 ARCHITECTURE 12

4.5.1 FLOWCHART 17

5.2.1 PHASES OF GSO 20

6.2.1 OUTPUT 28

6.2.2 GRAPHS 31

**vi**

**LIST OF ABBREVITIONS**

**MANET** Mobile Adhoc Network

**IDS**  Intrusion detection System

**ABID**  Anomaly Based Intrusion Detection

**MNB** Multinomial Naive Bayesian

**GSO** Glowworm Swarm Optimization

**vii**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **TABLE OF CONTENTS** | | | | |  | | |
| **CHAPTERNO** | **TITLE** | | | | | **PAGE NO** | | |
|  | **ABSTRACT** | | | | | **V** | | |
| **1**  **2**  **3**  **4**  **5**  **6**  **7**  **8** | **LIST OF FIGURES**  **LIST OF ABBREVIATIONS**  **INTRODUCTION**  **1.1.OVERVIEW**  **1.2 ATTACKS IN MANET**  **1.3.INTRUSION DETECTION SYSTEM**  **1.4.TYPES OF INTRUSION DETECTION SYSTEMS**  **LITERATURE SURVEY**  **SYSTEM ANALYSIS**  **3.1.EXISTING SYSTEM**  **3.2.PROPOSED SYSTEM**  **3.3.SYSTEM ARCHITECTURE**  **SYSTEM REQUIREMENTS**  **4.1.HARDWARE REQUIREMENTS**  **4.2.SOFTWARE REQUIREMENTS**  **4.3.SOFTWARE DESCRIPTION**  **4.4.TOOL COMMAND LANGUAGE**  **4.5.FLOWCHART**  **viii**  **MODULES DESCRIPTION**  **5.1.CLASSIFICATION**  **5.2.OPTIMAL PATH SELECTION**  **IMPLEMENTATION**  **6.1.SAMPLE CODING**  **6.2.RESULT**  **6.2.1.OUTPUT**  **6.2.2.GRAPHS**  **CONCLUSION AND FUTURE WORK**  **REFERENCES** | | | | | **VI**  **VII**  **1**  **4**  **5**  **8**  **11**  **18**  **21**  **34**  **35** | | |
|  |  | | | | |  | | |
|  |  | | | | |  | | |
|  |  | | | |  |  | | |
|  |  | | | |  |  | | |
|  |  | | | |  |  | | |
|  |  | | | |  |  | | |
|  |  | | | | **ix** |  | | |
|  |  | | | |  |  | | |
|  |  | | | | |  | | |
|  |  |  | | | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | | | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | | | |  | | |
|  |  | |  | | |  | | |
|  |  | |  | | |  | | |
|  |  | |  | | |  | | |
|  |  | | | | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | | |
|  |  | | |  | |  | |
|  |  | | |  | |  |
|  |  | | |  | |  |
|  |  | | |  | |  |
|  |  | | |  | |  |
|  |  | | |  | |  |
|  |  | | |  | |  |

**CHAPTER 1**

**INTRODUCTION**

* 1. **OVERVIEW**

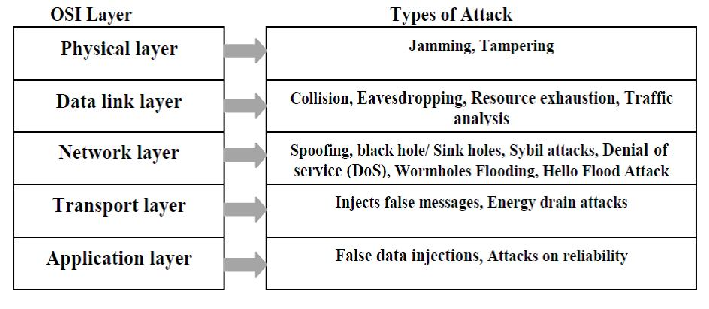
With the increasing use of handheld wireless devices(tablet computers, cell phones, mobile Internet devices, PDAs and so on) and recent advancement in wireless communication technology, Mobile Ad hoc Network(MANET) is gaining more importance in commercial, openness of MANET make it attractive for various types of application, such as military communication, emergency search and rescue operations, disaster recovery, fire fighting, policing, communication between moving vehicles (VANET), sensor networks, battlefields, conferences and so on. MANET is wireless local area network model without any centralized infrastructure such as base stations or access points. In order to provide the valid communication between two mobile nodes beyond their direct transmission range in MANETs, the intermediate nodes are used to forward the packets in multi-hop fashion. The basic assumption of all ad hoc routing protocols is that all mobile nodes would be reliable, trustworthy, and cooperative in basic operation of the network. However, MANETs are more

vulnerable to various types of security attacks because of their inherent characteristics, such as dynamic topology, multi-hop environment, error prone communication media, limited bandwidth, computing power constraints, limited physical security, and frequent routing updates. So, providing the secure communication over the MANETs is a major concern.

**1.2 ATTACKS IN MANET**

In MANETs, there is no clearly defined central place, where any security mechanism (to detect, prevent, or recover from security attack) can be deployed. Therefore, the security attacks from both external and internal nodes can compromise the security and privacy of the network. These attacks can freeze the entire operation of the network and violate the core security principles, i.e. confidentiality, integrity and availability.

The security attacks against MANETs are classified into two broad categories on the basis of nature: Passive attacks and Active attacks. Passive attacks are mainly against data confidentiality. In passive attacks, the unauthorized nodes attempt to break the system solely on the basis of monitoring the transmission over the channel without causing any damage to the network directly. They extract the valuable information and exploit it for future attacks. These attacks are very difficult to detect because they do not perform any direct damage. Active attacks are mostly against data confidentiality data integrity. Active attacks involve modifying, dropping, fabricating, replaying, or blocking the data packets from being exchanged in the network. These attacks are generally launched by authorized nodes of the network. They use the different features of the network to launch the attacks.

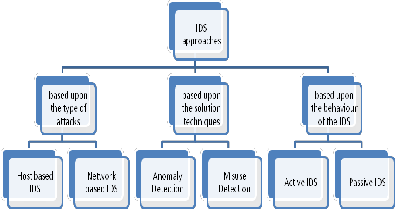
****

**Fig.1.2.1**

**1.3 INTRUSION DETECTION SYSTEM**

The IDS is the best security mechanism in the battle against the security attacks at various levels. Intrusion detection “is a process of monitoring the events occurring in a system or network, analyzing them for signs of possible incidents which represent a violation of

security policy and standards, and report unauthorized and malicious activities accordingly.” The IDS is a software and/or hardware entity to automate the detection of abnormal activities that attempt to compromise the integrity, confidentiality, or availability of a system with the following functionality.



**Fig.1.3.1**

**1.4 TYPES OF INTRUSION DETECTION SYSTEMS**

**Network Intrusion Detection Systems (NIDS**) usually consists of a network appliance (or sensor) with a Network Interface Card (NIC) operating in promiscuous mode and a separate management interface. The IDS is placed along a network segment or boundary and monitors all traffic on that segment.

**Host Intrusion Detection Systems (HIDS)** and software applications (agents) installed on workstations which are to be monitored. The agents monitor the operating system and write data to log files and/or trigger alarms. A host Intrusion detection systems (HIDS) can only monitor the individual workstations on which the agents are installed and it cannot monitor the entire network. Host based IDS systems are used to monitor any intrusion attempts on critical servers.

**A knowledge-based (Signature-based) Intrusion Detection Systems** (IDS) references a database of previous attack signatures and known system vulnerabilities. The meaning of word signature, when we talk about Intrusion Detection Systems (IDS) is recorded evidence of an intrusion or attack. Each intrusion leaves a footprint behind (e.g., nature of data packets, failed attempt to run an application, failed logins, file and folder access etc.). These footprints are called signatures and can be used to identify and prevent the same attacks in the future. Based on these signatures Knowledge-based (Signature-based) IDS identify intrusion attempts

**A Behavior-based (Anomaly-based) Intrusion Detection Systems** (IDS) references a baseline or learned pattern of normal system activity to identify active intrusion attempts. Deviations from this baseline or pattern cause an alarm to be triggered.

The anomaly detection technique centers on the concept of a baseline for network behavior. This baseline is a description of accepted network behavior, which is learned or specified by the network administrators, or both. Events in an anomaly detection engine are caused by any behaviors that fall outside the predefined or accepted model of behavior. An integral part of base lining network behavior is the engine’s ability to dissect protocols at all layers. For every protocol that is being monitored, the engine must possess the ability to decode and process the protocol in order to understand its goal and the payload. This protocol “dissection” is initially computationally expensive, but it allows the engine to scale as the rule set grows and alert with fewer false positives when variances from the accepted behaviors are detected. A disadvantage of anomaly-detection engines is the difficultly of defining rules. Each protocol being analyzed must be defined, implemented and tested for accuracy. The rule development process is also compounded by differences in vendor implementations of the various protocols. Custom protocols traversing the network cannot be analyzed without great effort. Moreover, detailed knowledge of normal network behavior must be constructed and transferred into the engine memory for detection to occur correctly.

On the other hand, once a protocol has been built and a behavior defined, the engine can scale more quickly and easily than the signature-based model because a new signature does not have to be created for every attack and potential variant. Another pitfall of anomaly detection is that malicious activity that falls within normal usage patterns is not detected. An activity such as directory traversal on a targeted vulnerable server, which complies with network protocol, easily goes unnoticed since it does not trigger any out-of-protocol, payload or bandwidth limitation flags.

In existing system trust factor and fuzzy based intrusion detection and prevention system is used for routing. Based on the trust values of the nodes, the fuzzy system identifies the intruder, such that the path generated in the MANET is secured. Moreover, an optimization algorithm, entitled Fuzzy integrated Particle Swarm Optimization (Fuzzy-FPSO), is proposed by the concatenation of the Firefly Algorithm (FA) and Particle Swarm Optimization (PSO) for the optimal path selection in order to provide secure routing. The traditional routing protocols can’t effectively predict and protect the MANET from various intruder attacks. One of the highly demanded aspects in MANET design is the evolution of a secure routing protocol doesn’t address several issues, like the prediction of selfish nodes in MANETs with high accuracy, reducing the computational overhead associated with the mechanism of cluster leader node selection. On implementation of SVM-based IDS, the packet delivery ratio decreases due to the availability of malicious nodes in the system. The end-to-end delay is maximum when malicious nodes attack the network. When the network is affected by a large number of malicious nodes, the effect becomes worse.

**CHAPTER 2**

**LITERATURE SURVEY**

1. **Farrukh Aslam Khan, Muhammad Imran, Haider Abbas, and Muhammad Hanif Durad, “A detection and prevention system against collaborative attacks in Mobile Ad hoc Networks,” Future Generation Computer Systems, vol. 68, pp. 416-427, 2017**

A Detection and Prevention System (DPS) technique to identify and thwart the malicious nodes in MANETs. For continuous monitoring of the nodal behavior some special nodes, named DPS nodes, were employed. When the DPS node had predicted some abnormal nodal behavior, the technique could assign that node as a wormhole, broadcasting the message to the other nodes to terminate the control and data messages to the particular node. The main advantage was it minimized the total number of dropped packets due to the malicious nodes with a minimum false positive rate. However, the technique is not suitable to modify the DPS system to prevent the network from other similar attacks.

**2. Erfan A Shams, and Ahmet Rizaner, “A Novel Support Vector Machine Based Intrusion Detection System For Mobile Ad Hoc Networks,” Wireless Networks, pp. 1-9, 2017.**

To predict the Denial of Service (DoS) type of attacks, a SVM-based IDS has developed. The system had provided continuous monitoring of the network to predict and eliminate the malicious nodes in the MANET, in order to enhance its performance. It had intensified the network reliability and was independent of the network size, node mobility and network routing protocol. The system could detect and remove the network attacks within a short computing time, and high detection rate, but the packet delivery ratio shrinks when a malicious node is available in the system.

**3. Usha, G., M. Rajesh Babu, and S. Saravana Kumar, “Dynamic anomaly detection using cross layer security in MANET,” Computers & Electrical Engineering, pp.1-11, 2016.**

A Honeypot Based Dynamic Anomaly Detection Using Cross-Layer Security (HBDADCS), for prediction and protection of the MANET from black hole attack. This technique enhanced the detection accuracy of attacks and provided better delivery of packets. However, due to the least percentage of black hole nodes, the network load is very low.

**4. B. Mukherjee, L.T. Heberlein, and K. N. Levitt, “Network Intrusion Detection”, IEEE Network Volume 8, Issue 3, May-June 1994**

Due to the widely used Internet and growing number of intrusions, intrusion detection systems (IDSs) have recently gained a considerable amount of interest. Intrusion detection systems (IDSs) “are based on the beliefs that an intruder’s behavior will be noticeably different from that of a legitimate user and that many unauthorized actions are detectable”. IDSs detect suspicious activities that may compromise system security and alert the systems administrator to respond to the threat. In fact, an IDS is not a preventive security system but more like an alarm system working together with other passive information assurance processes as an important element of defense in layers.

**5. Scarfone K, Mell P. Guide to intrusion detection and prevention systems (IDPS).Technical Report, National Institute of Standards and Technology (NIST) Special Publication 800-94.Department of Commerce, U.S., 2007.**

The IDS is the best security mechanism in the battle against the security attacks at various levels. A intrusion detection “is a process of monitoring the events occurring in a system or network, analyzing them for signs of possible incidents which represent a violation of security policy and standards, and report unauthorized and malicious activities accordingly.”

**6. F.Y.Schafire. A decision theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci. Vol.55, No.1, pp.119-139, 1997**

Ensemble of classifiers can often better than any individual classifier, which makes them an efficient approach in detecting network intrusions. The AdaBoost algorithm proposed by Y. Freund and R. Schapire is one of the most important ensemble methods, since it has solid theoretical foundation, very accurate prediction, great simplicity and very wide and successful applications

**7. Sung –Hae Jun and Kyung –whan oh, An evolutionary support vector machine for intrusion detection. Asian journal of information technology, Vol. 5, No. 7, pp. 778-783, 2006.**

An evolutionary support vector machine for intrusion detection is proposed in. In this, the authors have combined evolutionary programming into support vector machines. They concluded that their model is able to detect new attacks as well as experienced attacks.

**8.Nadeem, A., & Howarth, M. P. (2013). A survey of MANET intrusion detection & prevention approaches for network layer attacks. IEEE Communications Surveys & Tutorials, 15(4), 2027–2045.**

The distributed nature of MANETs means that it is vital to protect them from modern sophisticated network layer attacks. In this paper we have presented a survey of significant network layer attacks, and we have reviewed intrusion detection mechanisms that have been proposed in the literature. The protection mechanisms can be classified as either point detection algorithms or as IDSs that can deal with a wide range of attacks. In comparing the main proposals have highlighted a number of key similarities and differences between the various mechanisms.

9**. Amudhavel, J., Brindha, V., Anantharaj, B., Karthikeyan, P., Bhuvaneswari, B., Vasanthi, M., et al. (2016). A survey on intrusion detection system: State of the art review. Indian Journal of Science and Technology, 9(11), 1–9.**

The establishment of the status of feature decrease to typical insubstantial IDS. Finally, they suggest a hybrid style relating collective and improper classifiers for intrusion for detection. The first security level of intrusion detection is to exercise an intrusion detection system using an eminent traditional approach with the benefits of the above approaches. The second level of intrusion system detection is to state efficient security policies, in which the securities are based on the responsibilities.

10. **Poongodi, M., & Bose, S. (2015). A novel intrusion detection system based on trust evaluation to defend against DDoS attack in MANET. Arabian Journal for Science and Engineering, 40(12), 3583–3594**

The prevention mechanism against the DDoS attack has been proposed [11]. In this Firecol updated the score through which the potential attack has identified. DGSOT plays the vital role on clustering based on the score at the routing level. Such that secure aware routing of potential attack is possible. The work presented in this [12] paper concerns about filtering the false data in the routing nodes before it reaches the sink and saves the resources. The proposed solution involves using Advanced Encryption Standard (AES) by deploying keys to the mobile ad hoc nodes when the network is first deployed. The data are verified by usage of message authentication code

**CHAPTER 3**

**SYSTEM ANALYSIS**

**3.1 EXISTING SYSTEM**

The Intrusion detection system using trust factor and fuzzy based intrusion detection and prevention for routing. Based on the trust values of the nodes, the fuzzy system identifies the intruder, such that the path generated in the MANET is secured. Moreover, an optimization algorithm, entitled Fuzzy integrated Particle Swarm Optimization (Fuzzy-FPSO), is used by the concatenation of the Firefly Algorithm (FA) and Particle Swarm Optimization (PSO) for the optimal path selection in order to provide secure routing. The simulation of the existing methodology is NS2 simulator and analysis is carried out considering four cases, like without attack, flooding attacks, black hole attack and selective packet drop attack concerning throughput, delay and detection rate. All the secured feasible paths from source node to destination node are discovered based on trust computation, and the optimal path is chosen by employing the proposed FPSO having trust and distance as its objective function.

**DISADVANTAGES:**

* The method easily suffers from the partial optimism, which causes the less exact at the regulation of its speed and the direction.
* T he packet delivery ratio decreases due to the availability of malicious nodes in the system.
* The technique is not suitable to modify the DPS system to prevent the network from other similar attacks

**3.2 PROPOSED SYSTEM**

The intrusion Detection system using Glowworm swarm optimization with Multinomial Naive Bayesian Technique. The nodes which are initialized in the network is classified into trusted and malicious nodes using Multinomial Naive Bayesian classifier. This classifier provides a simple approach, with clear semantics, representing and learning probabilistic knowledge. After the prediction of the secured nodes and intruder nodes using the Naive Bayesian classifier, the next step is the generation of paths between the source and the destination through the trusted nodes. Glowworm Swarm optimization which is a type of nature-inspired, stochastic, meta-heuristic algorithm employed to deal with the hard optimization issues.GSO is used for the optimal path selection based on fitness value. The three evaluation metrics considered for the comparative performance analysis are throughput, delay ,packet delivery ratio and detection rate.

**ADVANTAGES**

* GSO algorithm to address the problem of pursuit of multiple mobile signal sources.
* Less complexity than fuzzy based classification.
* It improves the network security by minimizing the packet drop ratio, network overhead and normalized routing load

**3.3 SYSTEM ARCHITECTURE**

**FIG.3.3.1**

**CHAPTER 4**

**SYSTEM REQUIREMENTS**

**4.1 HARDWARE REQUIREMENTS**

* System : I3 2.53GHz VT Enabled
* Hard Disk : 500 GB
* RAM : 4GB

**4.2 SOFTWARE REQUIREMENTS**

* Operating system : Ubuntu 16.04
* Coding Language : Tool Command Language
* Package : NS2(Network Simulator)

**4.3 SOFTWARE DESCRIPTION**

NS2 stands for Network Simulator Version 2. It is an open-source event-driven simulator designed specifically for research in computer communication networks. NS2 is an open-source simulation tool that runs on Linux. It is a discreet event simulator targeted at networking research and provides substantial support for simulation of routing, multicast protocols and IP protocols, such as UDP, TCP, RTP and SRM over wired and wireless (local and satellite) networks. It has many advantages that make it a useful tool, such as support for multiple protocols and the capability of graphically detailing network traffic. Additionally, NS2 supports several algorithms in routing and queuing. LAN routing and broadcasts are part of routing algorithms. Queuing algorithms include fair queuing, deficit round-robin and FIFO.

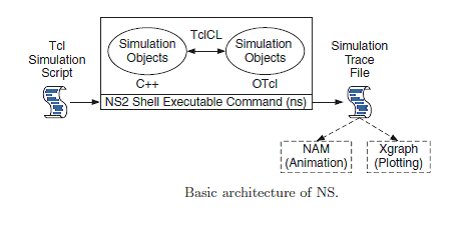


FIG.4.3.1

**NS2 Package Installation in UBUNTU**

1. **Download ns-allinone-2.34.tar.gz file** from

<http://sourceforge.net/projects/nsnam/files/>

1. **Copy the ns-allinone-2.34.tar.gz file in to the desktop directory.**

**Note:** All the commands listed from here assumes the ns2 to be in desktop; you can put the extracted file anywhere and change the address in the commands correspondingly.

1. **open terminal type**:

sudo apt-get update (to get list of update packages/libraries in OS)

1. **Install required libraries for ns2 using terminal**:

sudo apt-get install build-essential auto conf auto make libxmu-dev

sudo apt-get install xorg-dev g++ xgraph

1. **Extract the ns-allinone-2.34.tar.gz file by typing following commands in terminal:**

cd Desktop <Enter>

tar zxvf ns-allinone-2.34.tar.gz <Enter>

(A new folder named ns-allinone-2.34 will be generated.)

1. **To avoid installation errors modify the specified files in ns-allinone-2.34 folder as specified below:**

1. In ns-allinone-2.34/otcl-1.13/configure

change into

SHLIB\_LD=”gcc –shared”

instead of

SHLIB\_LD=”ld –shared”

1. ns-allinone-2.34/ns-2.34/tools/ranvar.cc line:219 change

return GammaRandomVariable::GammaRandomVariable(1.0 + alpha\_, beta\_).value() \* pow (u, 1.0 / alpha\_);

to

return GammaRandomVariable(1.0 + alpha\_, beta\_).value() \* pow (u, 1.0 / alpha\_);

1. Change the lines 183 % and 185 in file ns-allinone-2.34/ns-2.34/mobile/nakagami.cc to

resultPower = ErlangRandomVariable(Pr/m, int\_m).value();

and

resultPower = GammaRandomVariable(m, Pr/m).value(); respectively.

1. Next add a bellow line after line 64 in ns-allinone-2.34/ns-2.34/mac/mac-802\_11Ext.h

#include <stddef.h>

1. **Installation. In terminal goto Desktop/ns-allinone-2.34 and type:**

./install

This takes about 5min and note down the instructions you get for setting path.

1. **Setting Path**: According to instructions you get on the terminal after installation change following path
2. In terminal sudo gedit ~/.bash\_aliases

ii Then copy the following path into .bash\_aliases and modify according to your instructions:

# LD\_LIBRARY\_PATH

OTCL\_LIB=/home/ubuntu/Desktop/ns-allinone-2.34/otcl-1.13

NS2\_LIB=/home/ubuntu/Desktop/ns-allinone-2.34/lib

X11\_LIB=/usr/X11R6/lib

USR\_LOCAL\_LIB=/usr/local/lib

exportLD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:$OTCL\_LIB:$NS2\_LIB:$X11\_LIB:$USR\_LOCAL\_LIB

# TCL\_LIBRARY

TCL\_LIB=/home/ubuntu/Desktop/ns-allinone-2.34/tcl8.4.18/library

USR\_LIB=/usr/lib

export TCL\_LIBRARY=$TCL\_LIB:$USR\_LIB

# PATH

XGRAPH=/home/ubuntu/Desktop/ns-allinone-2.34/bin:/home/ubuntu/Desktop/ns-allinone- 2.34/tcl8.4.18/unix:/home/ubuntu/Desktop/ns-allinone-2.34/tk8.4.18/unix

NS=/home/ubuntu/Desktop/ns-allinone-2.34/ns-2.34/

NAM=/home/ubuntu/Desktop/ns-allinone-2.34/nam-1.14/

PATH=$PATH:$XGRAPH:$NS:$NAM

1. Then it takes effect immediately you should give below mentioned using terminal.

Source ~/. Bashrc

1. Now move to Desktop/ns-allinone-2.34/ns-2.34 and type:

./validate

Validations tests are performed for about 10-15 min.

10.If you type ns in your terminal % should be displayed.

**4.4 TCL(TOOL COMMAND LANGUAGE):**

**TCL (**Tool Command Language) is a scripting language created by John Ousterhout. A special simulator object is defined in NS2 that perform process-based simulation on the network objects. Tcl gained acceptance on its own. It is commonly used for rapid prototyping, scripted applications, GUIs and testing. Tcl is used on embedded systems platforms. Tcl's features include

* All operations are commands, including language structures. They are written in prefix notation. Commands are commonly variadic.
* Everything can be dynamically redefined and overridden. All data types can be manipulated as strings, including source code.
* Fully dynamic, class-based object system, TclOO, including advanced features such as meta-classes, filters, and mixins.
* Event-driven interface to sockets and files. Time-based and user-defined events are also possible.
* Variable visibility restricted to lexical (static) scope by default, but allowing process to interact with the enclosing functions' scopes.
* All commands defined by Tcl itself generate error messages on incorrect usage.

Extensibility, via C, C++, Java, and Tcl Interpreted language.

**4.5 FLOW CHART**

**Malicious Nodes**

**FIG.4.4.1**

**CHAPTER 5**

**MODULEs DESCRIPTION**

The primitive goal of this work is to construct a contemporary highly secure routing protocol by implementing trust factor and Naive Bayesian based IDPS methodology. For the Intrusion Detection and Prevention, here a Glowworm Swarm Optimization (GSO) algorithm is employed for predicting the secure route in the MANET. The four major phases of this research are i) Trust computation of the nodes, ii) Intrusion detection using Naive Bayesian classifier, iii) Path generation, and iv) Selection of the secured path using the proposed Glowworm Swarm Algorithm. In phase 1, the trust value of every node will be calculated on the basis of various trust factors. After the successful computation of trust value of nodes, the intrusion detection will be identified on the basis of Naive Bayesian classifier. The attackers intruding the network can be predicted by employing the proposed IDPS with Naive Bayesian along with trust factors. Once the secure nodes are identified, the newly designed Glowworm Swarm Optimization algorithm plays the vital role in the optimal path selection for secure routing.

**Input phase**- Nodes are connected with each other for relaying messages in mobile ad-hoc network. Every node is initialized with Trust index=0.5.

**Classifier**-the naive Bayesian classifier is used classifies the nodes into trusted and malicious nodes based on the trust values of the nodes.

**Routing protocol**- **Destination-Sequenced Distance-Vector Routing (DSDV)** is a table-driven routing scheme for ad hoc mobile network based on the Bellman–Ford algorithm which is used to exchange of messages between source and destination in a network

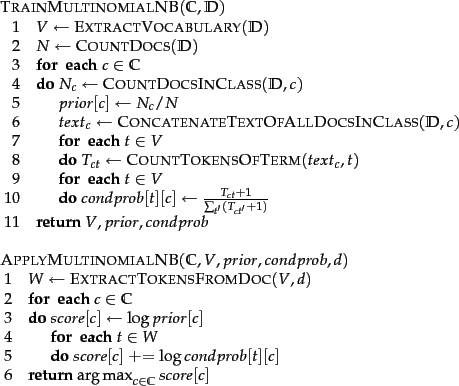
**Optimization algorithm**-Glowworm swarm optimization algorithm is used to identify the more suitable path for data transmission between source and destination among several paths generated using trusted nodes. This algorithm work on the basis of fitness value of the each path.

**Output phase**-In the output phase, The solution with the maximum fitness value is considered as the best solution for secure communication of two nodes.

**5.1 CLASSIFICATION:**

**Multinomial naive Bayes** classifier is used to recognize the nodes as trusted and malicious ones. Multinomial naive Bayes or MNB for short—generally outperforms the multivariate one and has also been found to compare favourably with more specialized event models . However, it is still inferior to the state-of-the-art support vector machine classifiers in terms of classification accuracy when applied to node categorization problems .It is easy to implement and has good running time and is claimed to be nearly as accurate as support vector machines.

ALGORITHM:

****

**5.2 OPTIMAL PATH SELECTION:**

Once the secure nodes are identified, the Glow worm Swarm optimization algorithm plays the vital role in the optimal path selection for secure routing. In GSO algorithm, the glow-worm is more attractive when the luciferin value is greater, which guides other glowworms to move towards it. Each glow worm has its dynamic decision space, which contains glowworms with both values of luciferin higher than itself and distance within its dynamic decision radius. Glowworm updates its location to a glow worm in its dynamic decision space in the light of probability, and then, renews its decision space radius. Here the luciferin value is considered as the fitness value of the each path which is generated between source and destination.

**FIG.5.2.1**

Initialization: Random distribution of nodes in the Network.

Optimal path generation

Neighbourhood update phase

Classification of nodes

Movement phase: node i move towards node j

Fitness value update phase: (previous fitness value + current value)

**CHAPTER 6**

**IMPLEMENTATION**

**6.1 SAMPLE CODING**

# Relay Selection

set val(chan) Channel/WirelessChannel ;# channel type

set val(prop) Propagation/TwoRayGround ;# radio-propagation model

set val(netif) Phy/WirelessPhy ;# network interface type

set val(mac) Mac/802\_11 ;# MAC type

set val(ifq) Queue/DropTail/PriQueue ;# interface queue type

set val(ll) LL ;# link layer type

set val(ant) Antenna/OmniAntenna ;# antenna model

set val(ifqlen) 50 ;# max packet in ifq

set val(nn) 21 ;# number of mobile nodes

set val(rp) DSDV ;# routing protocol

set val(x) 1030 ;# X dimension of topography

set val(y) 603 ;# Y dimension of topography

set val(stop) 20.0 ;# time of simulation end

#Create a ns simulator

set ns [new Simulator]

#Setup topography object

set topo [new Topography]

$topo load\_flatgrid $val(x) $val(y)

create-god $val(nn)

#Open the NS trace file

set tracefile [open relay.tr w]

$ns trace-all $tracefile

#Open the NAM trace file

set namfile [open relay.nam w]

$ns namtrace-all $namfile

$ns node-config -adhocRouting $val(rp) \

-llType $val(ll) \

-macType $val(mac) \

-ifqType $val(ifq) \

-ifqLen $val(ifqlen) \

-antType $val(ant) \

-propType $val(prop) \

-phyType $val(netif) \

-channel $chan \

-topoInstance $topo \

-agentTrace ON \

-routerTrace ON \

-macTrace ON \

-movementTrace ON

#Create 22 nodes

set n0 [$ns node]

$n0 set X\_ 354

$n0 set Y\_ 347

$n0 set Z\_ 0.0

$ns initial\_node\_pos $n0 20

set n1 [$ns node]

$n1 set X\_ 317

$n1 set Y\_ 432

$n1 set Z\_ 0.0

$ns initial\_node\_pos $n1 20

set n2 [$ns node]

$n2 set X\_ 243

$n2 set Y\_ 358

$n2 set Z\_ 0.0

$ns initial\_node\_pos $n2 20

set n3 [$ns node]

$n3 set X\_ 278

$n3 set Y\_ 265

$n3 set Z\_ 0.0

$ns initial\_node\_pos $n3 20

set n4 [$ns node]

$n4 set X\_ 406

$n4 set Y\_ 245

$n4 set Z\_ 0.0

$ns initial\_node\_pos $n4 20

set n5 [$ns node]

$n5 set X\_ 428

$n5 set Y\_ 440

$n5 set Z\_ 0.0

$ns initial\_node\_pos $n5 20

set n6 [$ns node]

$n6 set X\_ 463

$n6 set Y\_ 344

$n6 set Z\_ 0.0

$ns initial\_node\_pos $n6 20

set n7 [$ns node]

$n7 set X\_ 515

$n7 set Y\_ 250

$n7 set Z\_ 0.0

$ns initial\_node\_pos $n7 20

set n8 [$ns node]

$n8 set X\_ 523

$n8 set Y\_ 414

$n8 set Z\_ 0.0

$ns initial\_node\_pos $n8 20

set n9 [$ns node]

$n9 set X\_ 565

$n9 set Y\_ 325

$n9 set Z\_ 0.0

$ns initial\_node\_pos $n9 20

set n10 [$ns node]

$n10 set X\_ 604

$n10 set Y\_ 244

$n10 set Z\_ 0.0

$ns initial\_node\_pos $n10 20

set n11 [$ns node]

$n11 set X\_ 662

$n11 set Y\_ 328

$n11 set Z\_ 0.0

$ns initial\_node\_pos $n11 20

set n12 [$ns node]

$n12 set X\_ 649

$n12 set Y\_ 421

$n12 set Z\_ 0.0

$ns initial\_node\_pos $n12 20

set n13 [$ns node]

$n13 set X\_ 753

$n13 set Y\_ 357

$n13 set Z\_ 0.0

$ns initial\_node\_pos $n13 20

set n14 [$ns node]

$n14 set X\_ 754

$n14 set Y\_ 434

$n14 set Z\_ 0.0

$ns initial\_node\_pos $n14 20

set n15 [$ns node]

$n15 set X\_ 749

$n15 set Y\_ 503

$n15 set Z\_ 0.0

$ns initial\_node\_pos $n15 20

set n16 [$ns node]

$n16 set X\_ 848

$n16 set Y\_ 501

$n16 set Z\_ 0.0

$ns initial\_node\_pos $n16 20

set n17 [$ns node]

$n17 set X\_ 857

$n17 set Y\_ 436

$n17 set Z\_ 0.0

$ns initial\_node\_pos $n17 20

set n18 [$ns node]

$n18 set X\_ 930

$n18 set Y\_ 443

$n18 set Z\_ 0.0

$ns initial\_node\_pos $n18 20

set n19 [$ns node]

$n19 set X\_ 889

$n19 set Y\_ 348

$n19 set Z\_ 0.0

$ns initial\_node\_pos $n19 20

set n20 [$ns node]

$n20 set X\_ 826

$n20 set Y\_ 296

$n20 set Z\_ 0.0

$ns initial\_node\_pos $n20 20

$ns at 7.9 "$n7 add-mark n7 blue square"

$ns at 7.9 "$n8 add-mark n8 blue square"

$ns at 4.0 "$n9 add-mark n9 pink square"

$ns at 4.0 "$n1 add-mark n1 pink square"

$ns at 5.5 "$n0 add-mark n0 red square"

$ns at 7.9 "$n10 add-mark n10 blue square"

$ns at 7.9 "$n11 add-mark n11 red circle"

$ns at 7.9 "$n12 add-mark n12 red circle"

$ns at 3.0 "$n1 setdest 364.0 395.0 3.0"

$ns at 3.0 "$n2 setdest 121.0 320.0 3.0"

$ns at 3.0 "$n3 setdest 271.0 340.0 3.0"

$ns at 3.0 "$n4 setdest 236.0 382.0 3.0"

$ns at 3.0 "$n5 setdest 234.0 275.0 3.0"

$ns at 3.0 "$n7 setdest 234.0 175.0 3.0"

$ns at 3.0 "$n8 setdest 135.0 282.0 3.0"

$ns at 3.0 "$n9 setdest 197.0 195.0 3.0"

$ns at 3.0 "$n10 setdest 153.0 175.0 3.0"

$ns at 3.0 "$n11 setdest 334.0 275.0 3.0"

$ns at 3.0 "$n12 setdest 301.0 232.0 3.0"

$ns at 3.0 "$n13 setdest 264.0 425.0 3.0"

$ns at 3.0 "$n14 setdest 294.0 242.0 3.0"

$ns at 3.0 "$n15 setdest 264.0 225.0 3.0"

$ns at 3.0 "$n16 setdest 220.0 292.0 3.0"

$ns at 3.0 "$n17 setdest 197.0 240.0 3.0"

$ns at 3.0 "$n18 setdest 222.0 374.0 3.0"

$ns at 7.9 "$n8 label T"

$ns at 7.9 "$n7 label T"

$ns at 4.0 "$n9 label source"

$ns at 4.0 "$n1 label Destination"

$ns at 4.9 "$n9 label Hello"

$ns at 9.2 "$n9 label Hello"

$ns at 5.5 "$n0 label Attacker"

$ns at 5.9 "$n0 label Hello"

$ns at 6.9 "$n3 label Fine"

$ns at 7.2 "$n11 label A"

$ns at 7.9 "$n10 label T"

$ns at 7.2 "$n12 label A"

#Setup a TCP connection

set tcp1 [new Agent/TCP]

$ns attach-agent $n9 $tcp1

set sink1 [new Agent/TCPSink]

$ns attach-agent $n6 $sink1

set ftp1 [new Application/FTP]

$ftp1 attach-agent $tcp1

$ns at 4.1 "$ftp1 start"

$ns at 6.9 "$ftp1 stop"

#Setup a TCP connection

set tcp1 [new Agent/TCP]

$ns attach-agent $n6 $tcp1

set sink1 [new Agent/TCPSink]

$tcp1 set packetSize\_ 1500

#Setup a FTP Application over TCP connection

set ftp1 [new Application/FTP]

$ftp1 attach-agent $tcp1

$ns at 4.1 "$ftp1 start"

$ns at 6.9 "$ftp1 stop"

#Setup a TCP connection

set tcp1 [new Agent/TCP]

$ns attach-agent $n0 $tcp1

set sink1 [new Agent/TCPSink]

$ns attach-agent $n1 $sink1

$ns connect $tcp1 $sink1

$tcp1 set packetSize\_ 1500

$ftp1 attach-agent $tcp1

$ns at 4.1 "$ftp1 start"

$ns at 6.9 "$ftp1 stop"

#Setup a TCP connection

set tcp1 [new Agent/TCP]

$ns attach-agent $n0 $tcp1

set sink1 [new Agent/TCPSink]

$ns attach-agent $n3 $sink1

$ns connect $tcp1 $sink1

$ftp1 attach-agent $tcp1

$ns at 6.1 "$ftp1 start"

$ns at 7.0 "$ftp1 stop"

#Setup a TCP connection

set tcp1 [new Agent/TCP]

$ns attach-agent $n9 $tcp1

set sink1 [new Agent/TCPSink]

$ns attach-agent $n8 $sink1

$ns connect $tcp1 $sink1

$tcp1 set packetSize\_ 1500

set ftp1 [new Application/FTP]

$ftp1 attach-agent $tcp1

$ns at 8.4 "$ftp1 start"

$ns at 9.0 "$ftp1 stop"

#Setup a TCP connection

set tcp1 [new Agent/TCP]

$ns attach-agent $n8 $tcp1

set sink1 [new Agent/TCPSink]

$ns attach-agent $n1 $sink1

TCP connection

set ftp1 [new Application/FTP]

$ftp1 attach-agent $tcp1

$ns at 8.4 "$ftp1 start"

$ns at 9.0 "$ftp1 stop"

$ns at 0.0 "$ns trace-annotate \"Node creation...\""

$ns at 2.0 "$ns trace-annotate \"Network formation...\""

$ns at 4.0 "$ns trace-annotate \"Select the source and destination...\""

$ns at 4.9 "$ns trace-annotate \"send message to destination...\""

$ns at 5.5 "$ns trace-annotate \"attacker attack n0...\""

$ns at 5.9 "$ns trace-annotate \"Attacker modify the message content...\""

$ns at 7.0 "$ns trace-annotate \"Found trust node and Attacker node...\""

proc finish {} {

global ns tracefile namfile

$ns flush-trace

close $tracefile

close $namfile

exec xgraph delay &

exec xgraph pdr &

exec xgraph Throughput &

exec xgraph Throughput1

exit 0

}

for {set i 0} {$i < $val(nn) } { incr i } {

$ns at $val(stop) "\$n$i reset"

}

$ns at $val(stop) "$ns nam-end-wireless $val(stop)"

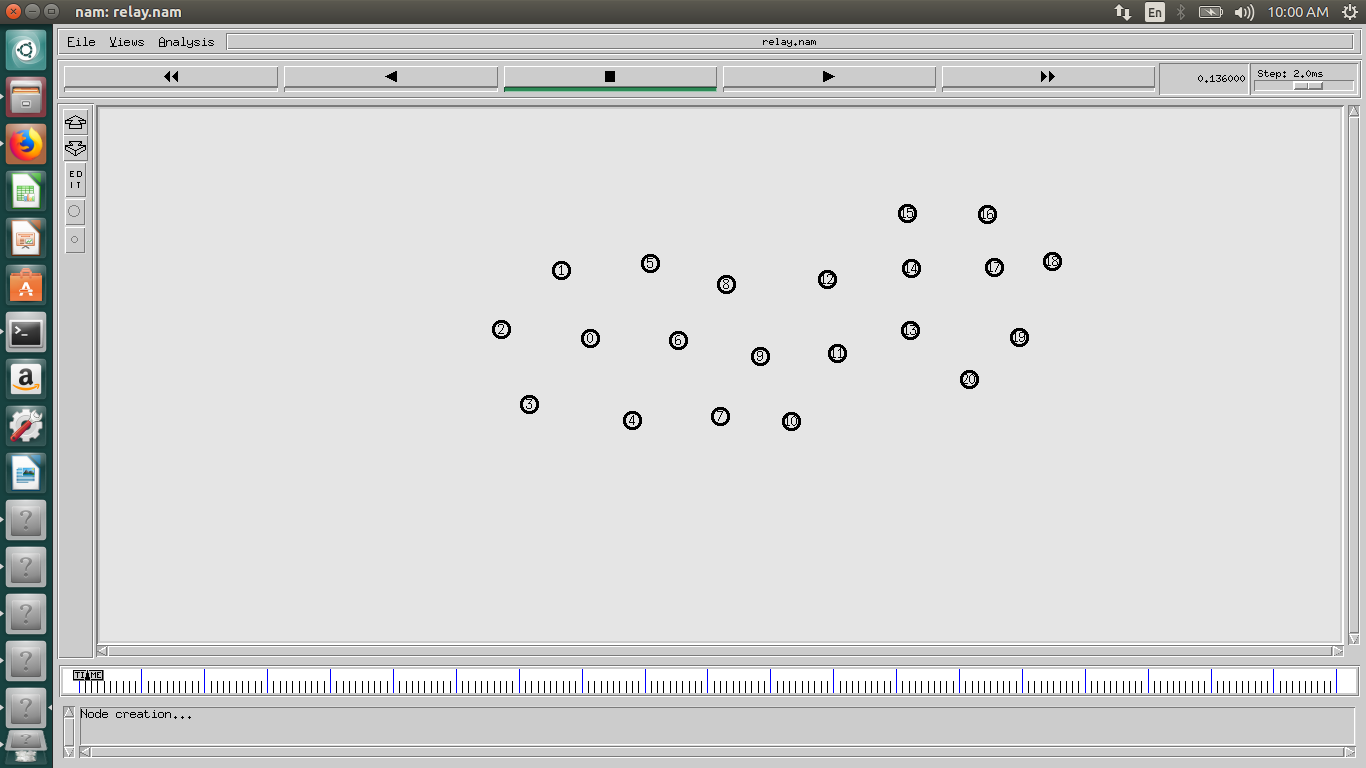
$ns at $val(stop) "finish"

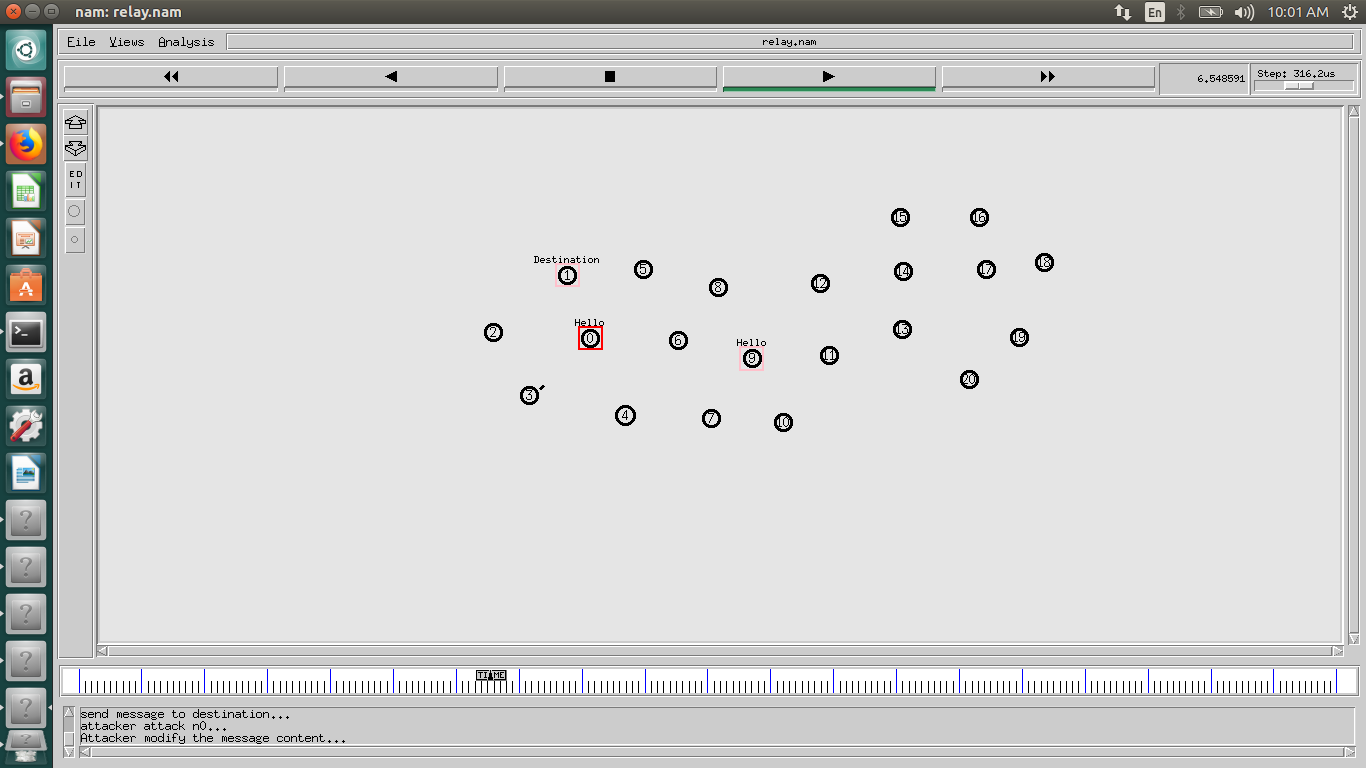
$ns at $val(stop) "puts \"done\" ; $ns halt"

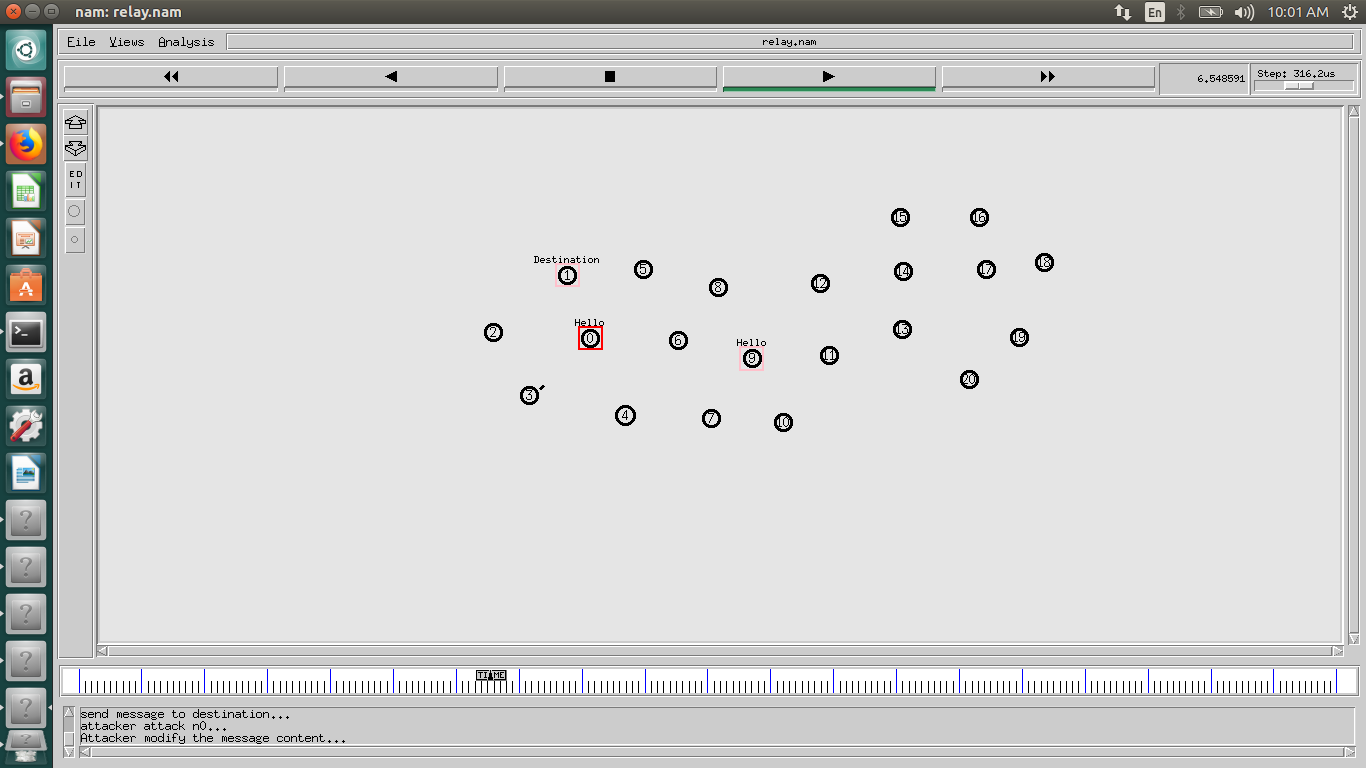
$ns run

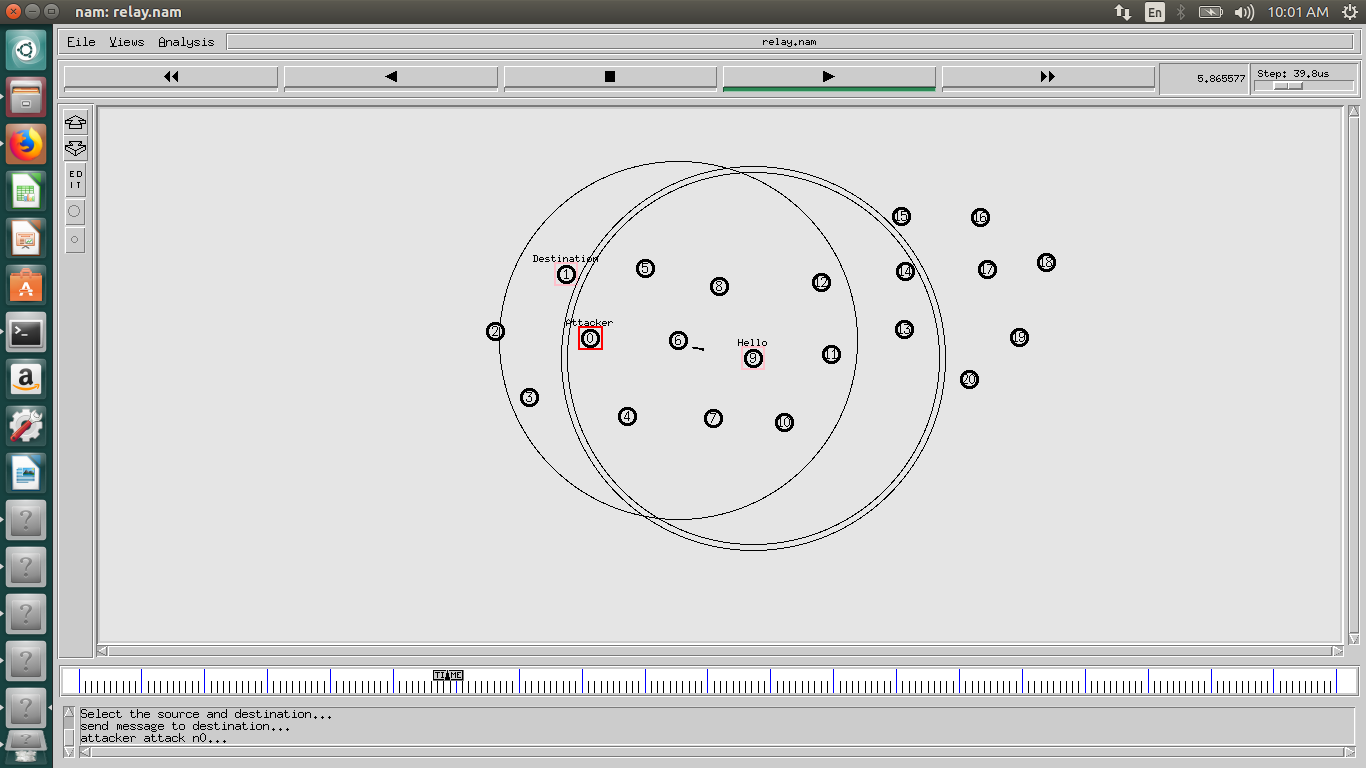
**6.2 RESULT:**

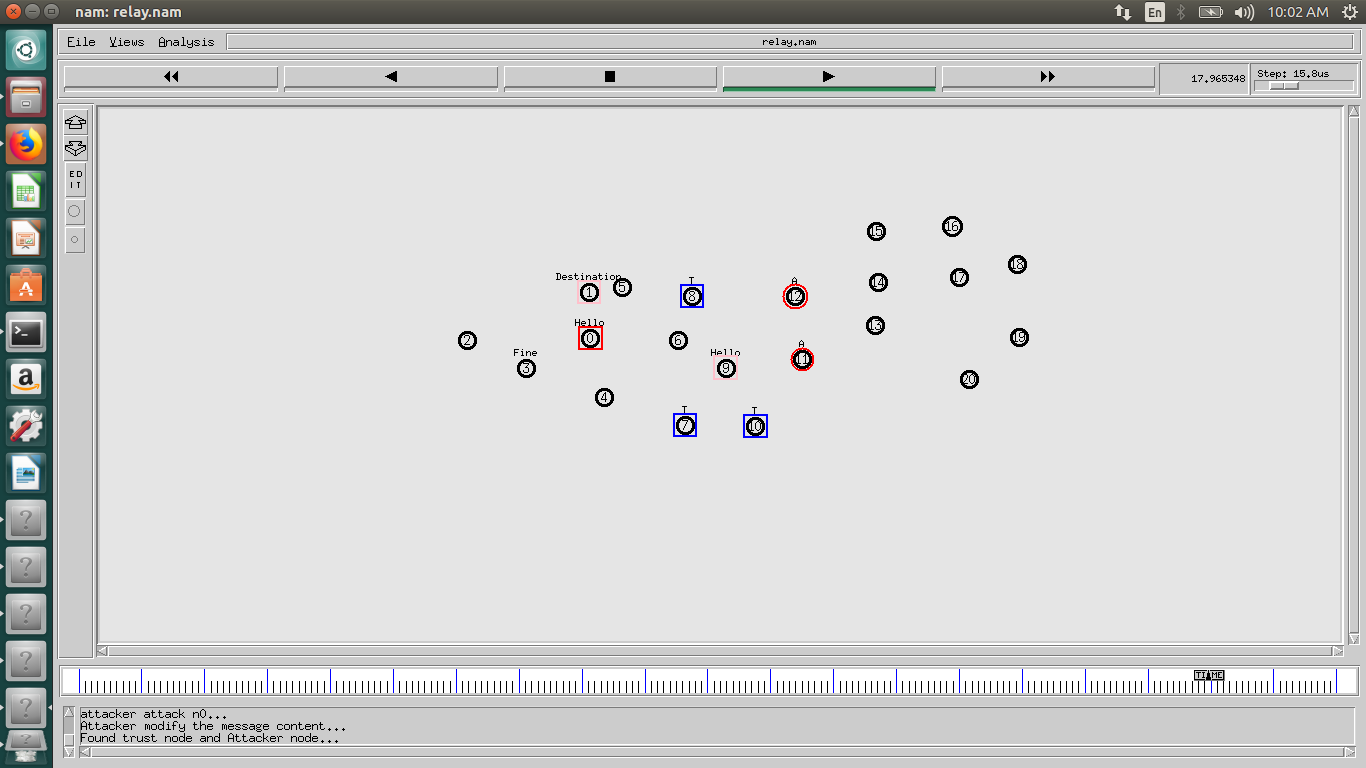
**6.2.1. OUTPUT:**

****

****

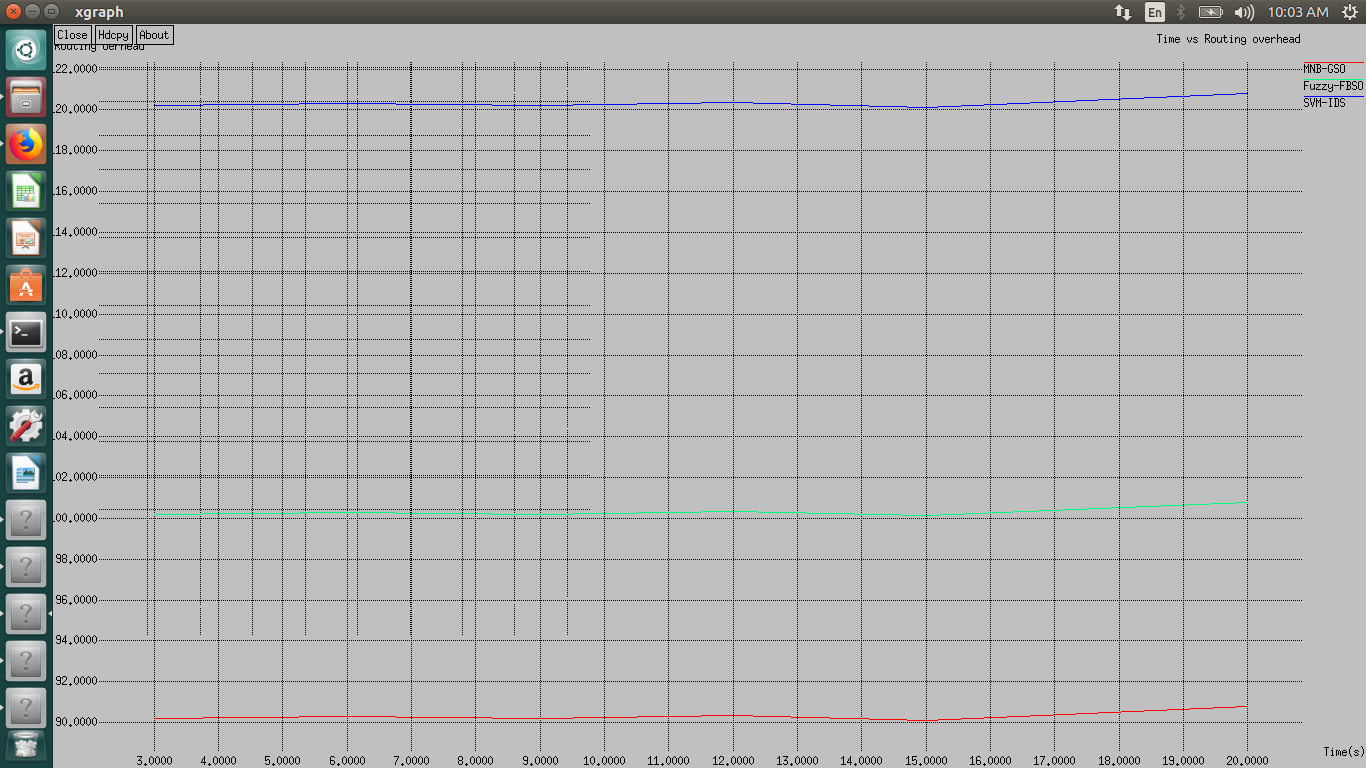
****

****

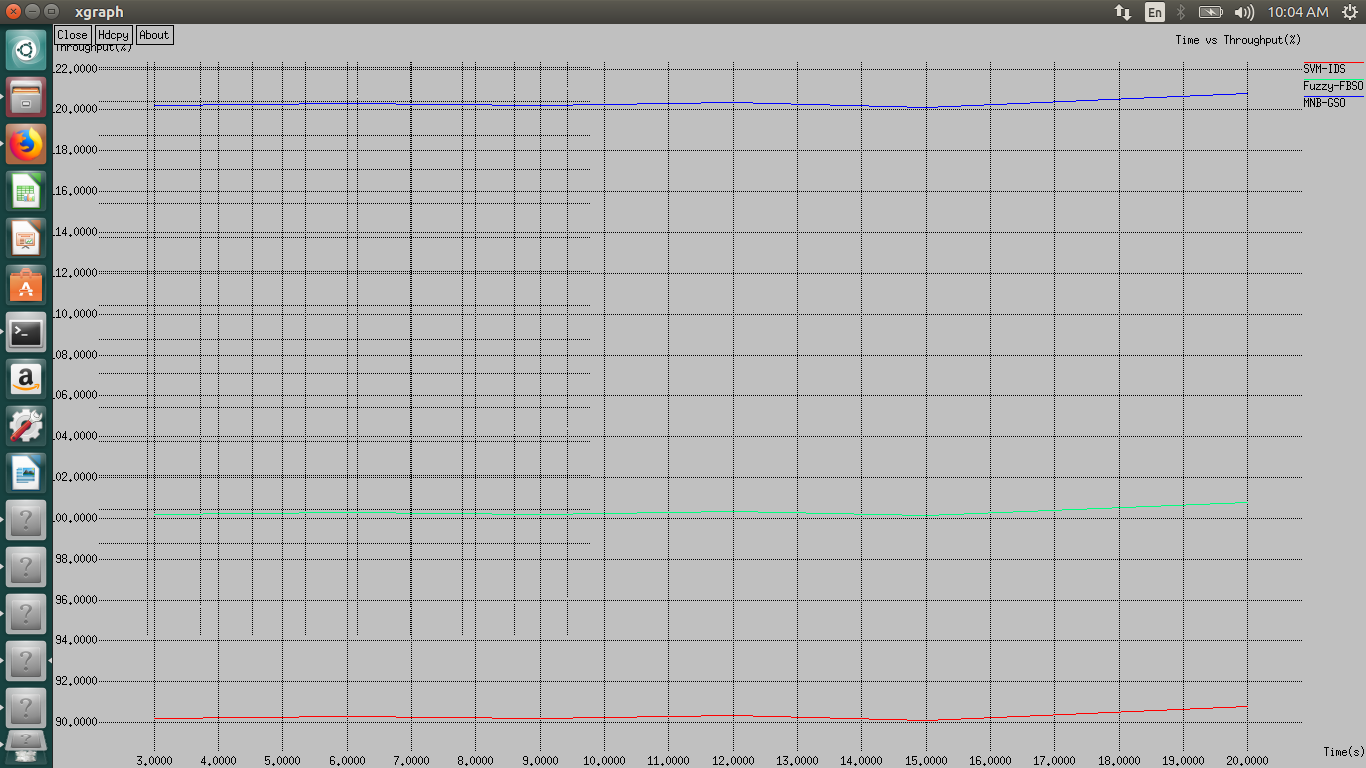
****

**6.2.2.GRAPHS**

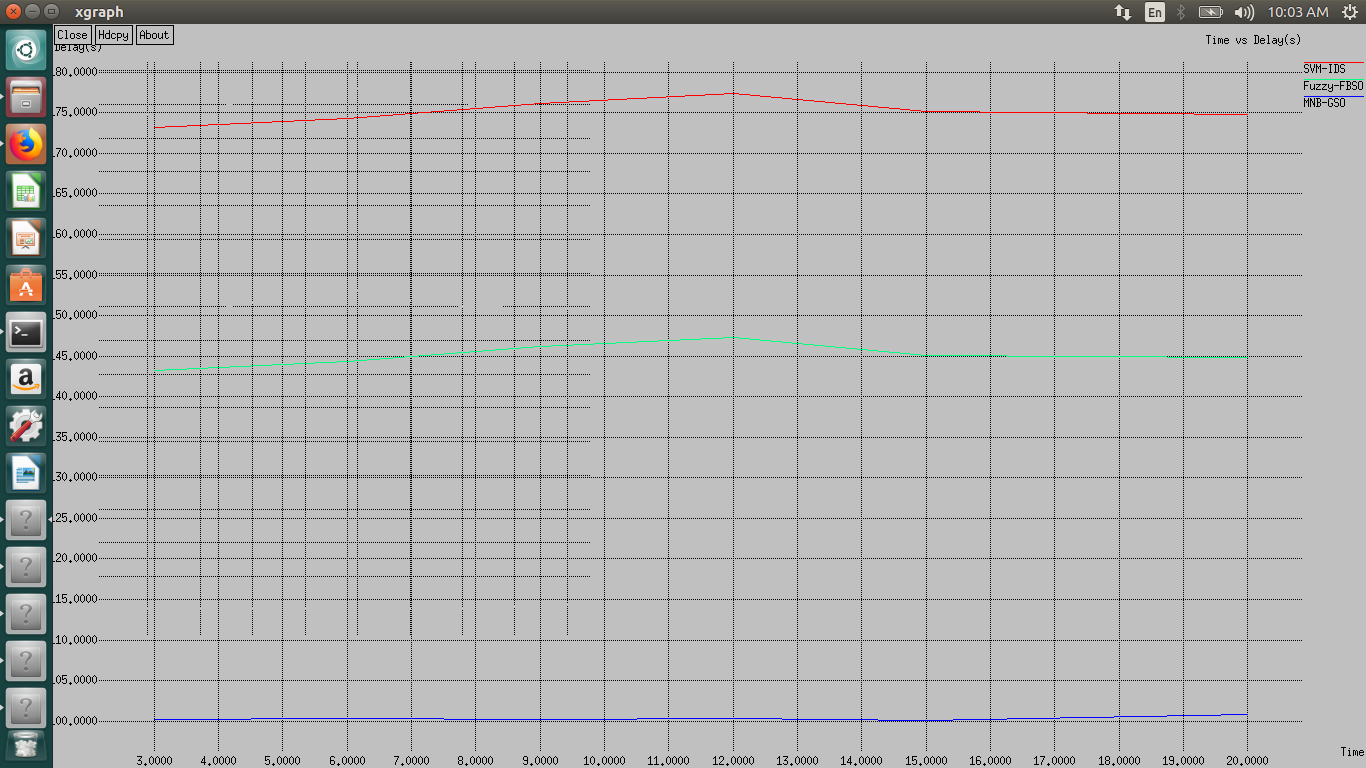
**6.2.2.1.ROUTING OVERHAED**

****

**6.2.2.2.THROUGHPUT**

****

**6.2.2.3.DELAY**

****

**CHAPTER 7**

**CONCLUSION & FUTURE WORK**

Security of MANET is one of the important features for its deployment, the detection and prevention of many attacks in the network exists as a challenging task. In this work analyzed the effect of black hole attack, selective packet dropping attack and flooding attack. The simulation has been done using the Network simulator (NS-2.35). The performance metrics like packet delivery ratio, overhead, throughput and average end to end delay has been measured and analyzed. From the simulation results it is clear that the existence of malicious nodes in the network, it also reduces the detection time and improved the effectiveness in detection process. It also reduces the computational overhead at nodes that leads to reducing in the message cost.

**FUTURE WORK:**

In future, we will combine different methods for Intrusion detection system and compare the performance with other. How different methods work in combination with the IDS and how detection time, accuracy and throughput get affected.

**CHAPTER 8**

**REFERENCES**

[1] Ramireddy Kondaiah1 and Bachala Sathyanarayana, ”Trust Factor and Fuzzy-Firefly Integrated Particle Swarm Optimization based Intrusion Detection and Prevention System for secure routing of MANET International Journal of Computer Networks & Communications (IJCNC) Vol.10, No.1, January 2018

[2] M. Nekovee, and R. S. Saksena,“Simulations of large-scale Wi-Fi-based wireless networks,” Interdisciplinary challenges and applications, Future Generation Computer Systems, vol. 26, no. 3, pp. 514–520, 2010.

[3] Poonam Joshi, Pooja Nande, Ashwini Pawar, Pooja Shinde, and Rupali Umbare, "EAACK-A Secure Intrusion Detection And Prevention System For MANETS," in proceedings of IEEE International Conference on Pervasive Computing (ICPC), pp. 1-6, 2015.

[4] Babu, M. Rajesh, and G. Usha, "A Novel Honeypot Based Detection and Isolation Approach (NHBADI) to Detect and Isolate Black Hole Attacks in MANET," Wireless Personal Communications, vol. 90, no. 2, pp. 831-845, 2016.

[5] arrukh Aslam Khan, Muhammad Imran, Haider Abbas, and Muhammad Hanif Durad, "A detection and prevention system against collaborative attacks in Mobile Ad hoc Networks," Future Generation Computer Systems, vol. 68, pp. 416-427, 2017

[6] Basant Subba , Santosh Biswas, and Sushanta Karmakar, "Intrusion detection in Mobile Ad-hoc Networks: Bayesian game formulation," Engineering Science and Technology, an International Journal, vol. 19, no.2, pp. 782-799, 2016.

[7] Marchang, Ningrinla, Raja Datta, and Sajal K. Das, "A Novel Approach for Efficient Usage of Intrusion Detection System in Mobile Ad Hoc Networks," IEEE Transactions on Vehicular Technology, vol. 66, no. 2, pp. 1684-1695, 2017.

[8] Usha, G., M. Rajesh Babu, and S. Saravana Kumar, "Dynamic anomaly detection using cross layer security in MANET," Computers & Electrical Engineering, pp.1-11, 2016.

[9] Erfan A Shams, and Ahmet Rizaner, "A Novel Support Vector Machine Based Intrusion Detection System For Mobile Ad Hoc Networks," Wireless Networks, pp. 1-9, 2017.

[10] Nadeem, Adnan, and Michael P. Howarth, "A survey of MANET intrusion detection & prevention approaches for network layer attacks," IEEE communications surveys & tutorials, vol. 15, no. 4, pp. 2027-2045, 2013.

[11] Opinder Singh, Jatinder Singh, and Ravinder Singh, "An Intelligent Intrusion Detection and Prevention System for Safeguard Mobile Adhoc Networks against Malicious Nodes," Indian Journal of Science and Technology, vol. 8, no. 1, pp. 1-12, 2017.

[12] Hamed Janzadeh, Kaveh Fayazbakhsh, Mehdi Dehghan, and Mehran S. Fallah, ”A secure credit based cooperation stimulating mechanism for MANETs using hash chains,” Future Generation Computer Systems, vol. 25, no. 8, pp. 926-934, 2009.

[13] Soni M, Ahirwa M, and Aggarwal S, “A Survey on Intrusion Detection in MANET,” in proceedings of International Conference on Computational Intelligence and Communication Networks, pp. 1027-1032, 2015.

[14] Meenakshi Patel, and Sanjay Sharma “Detection of Malicious Attack in MANET Behavioral Approach” 2013 3rd IEEE International Advance Computing Conference (IACC)

[15] Hizbullah Khattak and Nizamuddin, Fahad Khurshid,” Preventing Black and Gray Hole Attacks in AODV using Optimal Path Routing and Hash” 2013 IEEE.

[16] Seryvuth Tan, Keecheon Kim,” Secure Route Discovery for Preventing Black Hole Attacks on AODV-based MANETs” in 7/2013 IEEE.

[17] Ms Monika Y. Dangore and Mr Santosh S. Sambare,” Detecting And Overcoming Blackhole Attack In manet” 2013 International Conference on Cloud & Ubiquitous Computing & Emerging Technologies

[18] Ketan S. Chavda and Ashish V.Nimava,” REMOVAL OF BLACK HOLE ATTACK IN AODV ROUTING PROTOCOL OF MANET” 4th ICCCNT – 2013 July 4 -6, 2013, Tiruchengode, India

[19] jalpa khamar and avani Dadhania “An Performance Enhancement of AODV Routing Protocol in Manets”in IJAERD volume 1,Issue 6,june e-ISSN: 2348[10]. S. Abbas, M. Merabti, D. Llewellyn-Jones, and K.Kifayat,“Lightweight sybil attack detection in manets,” IEEE Syst. J.,vol. 7, no. 2, pp. 236–248, Jun. 2013.

[20]. J. Hortelano, J.-C. Cano, C. T. Calafate, M. de Leoni, P.Manzoni and M. Mecella, “Black hole attacks in p2p mobile networks discovered through Bayesian filters,” in Proc. Int.

Conf. Move Meaningful Internet Syst., 2010, pp. 543–552.