Chapter 1

INTRODUCTION

Due to widespread availability of mobile devices, a lot of research has gone into the development of MANETs and it is widely used. Mobile Ad Hoc Network(MANET) is also known as wireless Ad Hoc network or mesh mobile network. An ad hoc network is a network that is composed of individual devices communicating with each other directly. MANETs are self-configuring Ad Hoc networks i.e., its free to move independently in any direction and change its links to other mobile devices.
In MANET each node is able to send and receive the routing request, so it can act as host or router. Due to their infrastructure-less property and self-configuring nature, it poses serious drawbacks from security point of view, like routing attacks.
We focus on black-hole attacks which is a form of Denial of Service attack under routing attacks. Black-hole attack is an attack on routers where the malicious node sends a router reply informing that it has the shortest distance to the destination.

1.1 Purpose

The purpose of this project is to build a better security system for MANETs, since MANETs are used extensively. The security systems to prevent black-hole attacks either take a long time to detect the attack or have high calculation overhead. We propose a system which is simpler and more effective for the detection and prevention of black-hole attacks.

1.2 Scope

The research can be implemented for mission critical systems. And also for daily communication to make it more secure. More work can be gone into the research to make it more efficient and to be used for all other types of routing attacks.

1.3 Mobile Ad Hoc Networks(MANETs)

A wireless ad-hoc network is an accumulation of portable/semi-versatile hubs with no pre-built up foundation, shaping a brief network. Every one of the hub has a wireless interface and speaks with each other over either radio or infrared. PCs advanced collaborators that discuss specifically with each other are a few cases of hubs in an ad-hoc network. Hubs in the ad-hoc network are regularly portable, however can likewise comprise of stationary hubs, for example, get to focuses to the web[2].
Semi versatile hubs can be utilized to transfer focuses in territories where hand-off Figure 1.1 demonstrates a basic ad-hoc network with three hubs. The furthest hubs are not inside transmitter scope of each other. However the center hub can be utilized to forward bundles between the furthest hubs. The center hub is going about as a switch and the three hubs have framed an ad-hoc network.

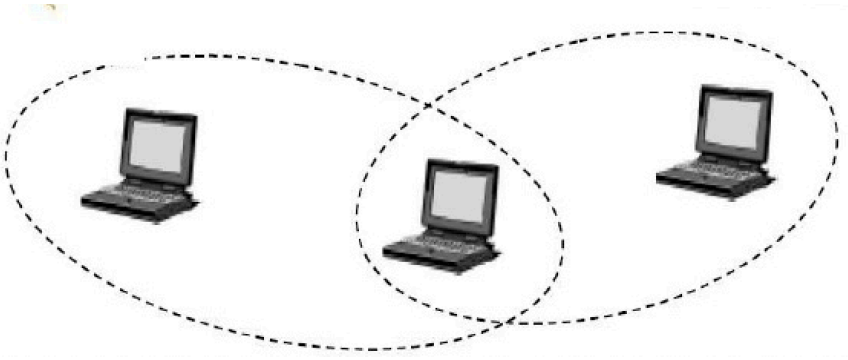


Fig 1.1: An Ad-Hoc Network

An ad-hoc network utilizes no concentrated administration. This is to make certain that the network won't fall since one of the portable hubs moves out of transmitter scope of the others. Hubs ought to have the capacity to enter/leave the network as they wish. In light of the restricted transmitter scope of the hubs, multi-bounce jumps might be expected to achieve different hubs. Each hub wishing to take an interest in ad-hoc network must will to forward bundles for different hubs. Accordingly every hub demonstrations both as a router, A hub can be seen as a conceptual element comprising of a router and an arrangement of associated portable hosts. A router is a substance, which, in addition to other things runs a steering convention. A portable host is just an IP-addressable host/element in the traditional sense.
Ad-hoc network are additionally equipped for taking care of topology changes and glitches in hubs. It is settled through network reconfiguration. For example, if a hub leaves the network and causes interface breakages, influenced hubs can without much of a stretch demand new routes and issue will be settled. This marginally expanded the deferral, yet the network will in any case be operational.

Wireless ad-hoc networks take advantages of the idea of the wireless correspondence medium. At the end of the day, in a wired network the physical cabling is completed an earlier limiting the association topology of the hubs. This confinement is absent in the wireless space and, gave that two hubs re inside transmitter scope of each other, a prompt between them may shape.

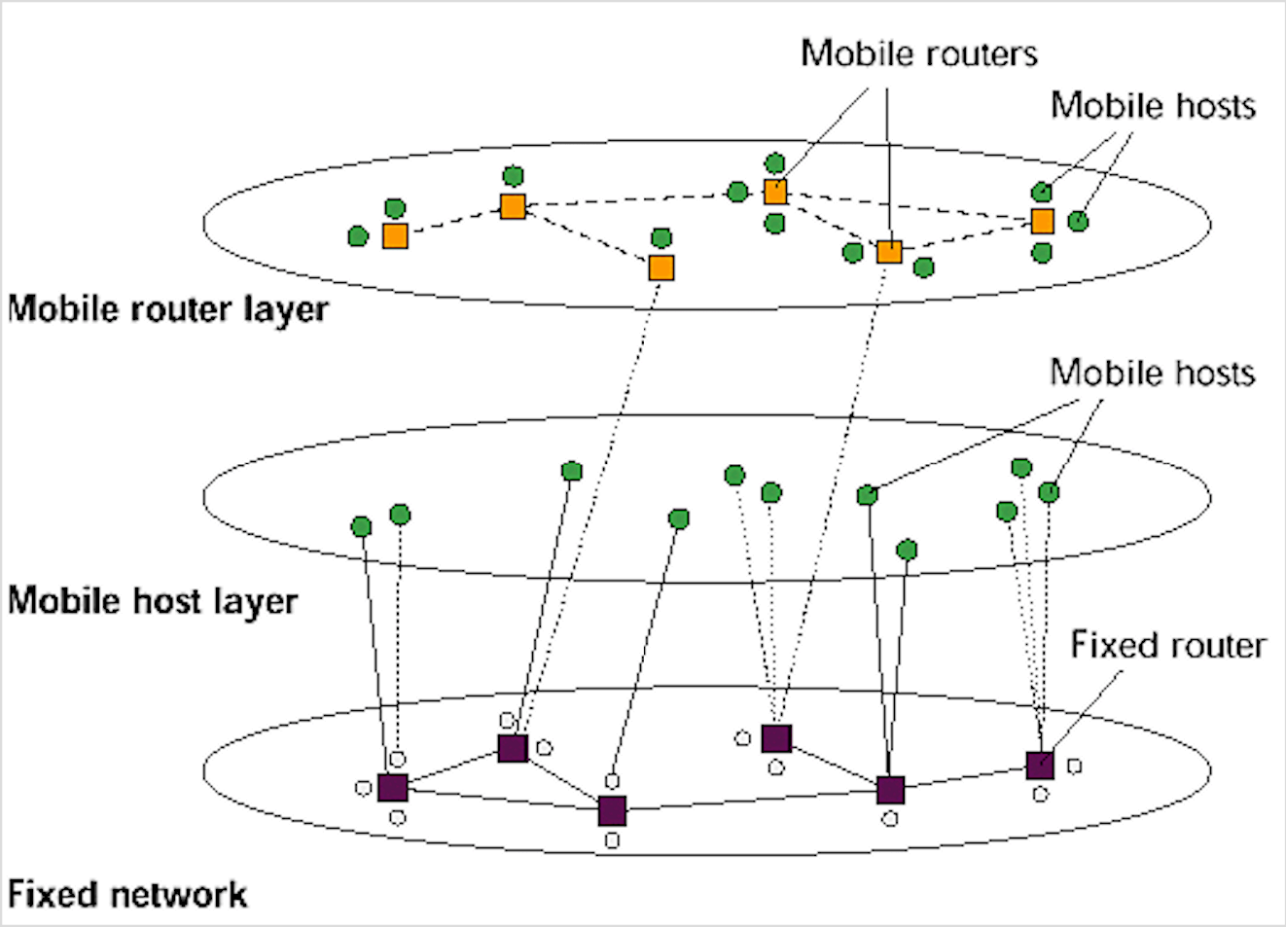


Fig 1.2: MANET Connection infrastructure

In the course of the most recent couple of years ad-hoc networking has pulled in a considerable measure of research intrigue. This has prompted formation of working gathering at the IETF that is concentrating on versatile ad-hoc networking, called MANET(Fig 1.2). Flexible IP and DHCP handle the relationship of the PDAs to settled system, MANET contains convenient IP for conveyability support and DHCP. MANETs and portable IP as a wellspring of numerous parameters, for example, an IP address.

MANET investigate is in charge of creating conventions and parts to empower ad-hoc networking between cell phones. It ought to be noticed that the detachment of end framework and router is just a sensible partition. Commonly, portable hubs in an ad-hoc situation involve steering and end frame-

work usefulness. The purpose behind having a unique area about ad-hoc networks inside a part about the network layer is that directing of information is a standout amongst the most troublesome issues in ad-hoc networks. A few cases for directing calculations suited to ad-hoc networks. NB: steering capacities now and again exist in layer 2, not simply in the network layer of the reference demonstrate. Bluetooth, for instance, offers sending/steering capacities in layer 2 in light of MAC addresses for ad-hoc networks. One of the main ad-hoc wireless networks was the parcel radio network. This made a simple association conceivable to the ARPA net, the beginning stage of the present web. Twenty radio channels between 1718-1840 MHz were utilized offering 100 or 400 kbit/s. The framework utilized DSSS with 128 or 32 chips/bit. A variation of separation vector directing was utilized as a part of this ad-hoc network. In this approach, every hub sends a directing advertisement each 7.5 s.

These ads contain an adjacent table with a rundown of connection characteristics to each neighbor. Every hub refreshes the neighborhood directing table as indicated by the separation vector calculation in view of these advertisements. Gotten bundles additionally refreshed the directing table. A sender now transmits a parcel to its first bounce neighbor table. A few improvements to this basic plan are expected to stay away from the steering circles and mirror the potentially quick evolving topology. The accompanying segments examine directing issues and improved steering instruments for ad-hoc networks in more detail. Perkins involves an accumulations of numerous directing conventions together with some underlying execution contemplations.

1.4 Definitions and Acronyms

- **PDR**: PDR is portrayed as the extent of data packs got by the objectives to those created by the sources. Numerically, it can be portrayed as: PDR= a1/a2 Where, a1 is the entire of data packs got by the each objective and a2 is the aggregate of data packages made by the each source.
- **Throughput**: Throughput is the most outrageous rate of creation or the best rate at which something can be prepared.
  Right when used as a piece of the setting of correspondence frameworks, for instance, Ethernet or bundle radio, throughput or framework throughput is the rate of productive message transport over a correspondence channel. The data these messages have a place with may be passed on finished a physical or sensible association, or it can experience a particular framework center point. Throughput is for the most part assessed in bits consistently, and occasionally in data distributes second or data packs per plan opportunity.
- **End-to-End delay**: End-to-end delay implies the time taken for a package to be transmitted over a framework from source to objective. It is a regular term in IP organize watching, and differs from round-trip time (RTT) in that selective path in the one course from source to objective is evaluated.
- **AODV**: Ad Hoc On-Demand Distance Vector (AODV) Routing is a coordinating convention for convenient off the cuff frameworks and distinctive remote offhand frameworks. It was commonly made in Nokia Research Center, University of California, Santa Barbara and University of Cincinnati by C. Perkins, E. Belding-Royer and S. Das.
  AODV is the directing convention utilized as a bit of ZigBee , a low information rate remote unrehearsed structure. There are unmistakable executions of AODV, for example, MAD-HOC, Kernel-AODV, AODV-UU, AODV-UCSB and AODV-UIUC.
  The AODV controlling convention is gotten ready for use by compact center points in an exceptionally selected framework. It offers rapid adjustment to dynamic connection conditions, low planning and memory overhead, low framework use, and chooses unicast courses to objectives inside the off the cuff framework. It uses objective gathering numbers to ensure circle adaptability constantly (notwithstanding despite anomalous conveyance of coordinating control messages), avoiding issues, related with traditional separation vector conventions.
- **Routing Overhead**:To keep up-to-date information about network routes, routing algorithms generate small sized packets, called routing packets. One example of such packets is a HELLO

packet, which is used to check whether the neighbour node is active. Note that routing packets do not carry any application content, like data packets do.
Both, routing and data packets have to share the same network bandwidth most of the times, and hence, routing packets are considered to be an overhead in the network. This overhead is called routing overhead. A good routing protocol should incur lesser routing overhead.

- **Black-hole attack**: In network arranging, a bundle drop assault or blackhole ambush is a kind of refusal of-benefit assault in which a hub that should hand-off packages rather arranges them off. This typically occurs from a hub getting the opportunity to be traded off from different assorted causes. One reason mentioned in explore is through a refusal of-benefit assault on the router utilizing a known DDoS tool. Since packages are routinely dropped from a lossy framework, the package drop assault is hard to recognize and check.
  The malignant router or hub can similarly accomplish this ambush particularly, e.g. by dropping packs for a particular framework objective, at a particular time, a package every n groups or every t seconds, or an erratically picked piece of the bundles. This is decently called a dark gap assault. If the toxic change endeavors to drop all packages that come in, the attack can truly be discovered nicely quick through typical frameworks administration instruments, for instance, traceroute. Additionally, when distinctive switches see that the traded off switch is dropping all development, they will overall begin to oust that change from their sending tables and at last no movement will stream to the attack. In any case, if the noxious switch begins dropping groups on a specific day and age or over every n bundles, it is habitually harder to distinguish on the grounds that some action still streams over the framework.

1.5 Literature Survey

1. "Detection and Removal of Cooperative Black and Gray Hole Attack in Mobile Ad-Hoc Networks"

This paper proposed a system to distinguish the malignant nide by isolating the total information movement into little estimated squares. The destination gets a block message from the source and sends a postlude message as an answer. The strategy is partitioned into two stages. To start with stage manages the information misfortune and second stage manages the detection of malevolent hub.
At first stage, with reception of postlude message, the source hub checks the information misfortune amid transmission of parcels is inside the limit run. If not, the second stage is started. The source hub sends a question message that incorporate day and age to every one of its neighbors, to recognize and expel a noxious hub.
At the point when a timeout strikes result in message or hub is malevolent, message answered to the source hub. The source hub at that point appends that hub in the findmalicious table and instates esteem voting as one on the off chance that it isn't already there and increases by one if the hub is already present in the findmalicious table.
In the event that that voting check surpasses the limit esteem, hub considered as a malevolent.

Limitations:

- This technique can be activated only when there is a real information transmission occurring.
- The plan requires additional preparing on source hub for separating movement and furthermore for the handling of precluding– postlude messages.
- The scheme is not efficient in cases of traffic congestion.

2. "Prevention of cooperative black hole attack in wireless ad hoc networks."

This paper proposed a methodology for identifying multiple black hole nodes cooperating as a group with the use of slightly modified AODV protocol by introducing Data Routing Information (DRI) Table and Cross Checking.
The solution involves two bits of additional information from the nodes responding to the RREQ of source node. In the DRI table, 1 denotes 'true' and 0 is for 'false'. The first bit "From" stands for information about routing data packet from the node (in the Node field) while the second bit "Through"

stands for information about routing data packet through the node (in the Node field).
Whenever an intermediate node (IN) responds to a RREQ, it sends the id of its next hop neighbor (NHN) and DRI entry for NHN to the source node. If IN is unreliable for the source then source sends a further route request (FREQ) to NHN. NHN in turn responds with a FREP message that includes DRI entry for IN, the next hop node of the current NHN, and the DRI entry for the current NHN's next hop.
If NHN is trusted node then source checks whether IN is a black hole or not using the DRI entry for IN replied by NHN and that for NHN replied by IN. They are consistent if IN is not malicious. If NHN is unreliable then the same cross checking will be continued with the next hop node of NHN. This cross checking loop continues until a trusted node is found.

Limitations:
The solution fails to accommodate the Grayhole Attack where the nodes keep alternating between malicious and normal behavior.

1.6 Existing Methods

1. 2ACK-Scheme

2ACK plan imperatively improves the location instrument Details of the 2ACK Scheme. The 2ACK plan is a system layer method to discover interfaces and to palliate their belongings. It can be executed as an extra to existing way conventions for MANETs, for example, OLSR and some other steering conventions. The 2ACK plan finds a decent conduct using another kind of affirmation package, named 2ACK[3].

A 2ACK package is appointed a settled way of two jumps (three hubs) in the opposite heading o the information movement way. The 2ACK plan is a system layer method to distinguish acting mischievously interfaces and to moderate their belongings. It can be actualized as an extra to existing directing conventions for MANETs, for example, OLSR. The 2ACK plan recognizes mischief using another sort of affirmation parcel, named 2ACK. A 2ACK bundle is appointed a settled course of two bounces (three hubs) the other way of the information movement course. It can be actualized as an extra to existing way conventions for MANETs, for example, OLSR and some other directing conventions.

The 2ACK plan finds a decent conduct using another kind of affirmation package, named 2ACK. A 2ACK package is appointed a settled way of two bounces (Fig 1.3) in the opposite heading of the information movement way. At N1, every ID will stay on the record for 't' seconds, the rest for 2ACK gathering. On the off chance that 2ACK groups coordinating to this ID touch base in front the clock leaves, the ID will be took out from the records. Other than, the ID will be taken out at the remainder of its post time partition and a counter called Cmis will be augmented. On the off chance that N3 gets an information package, at that point ascertained whether it needs to send a 2ACK package to N1. With a specific end goal to chop down the additional way overhead reason by the 2ACK framework, just a gap the information package will be recognized verses multi jump package. Such a gap named the affirmation extent, Rack. By evolving Rack, we can progressively tune up the overhead of Many-Hop package transmissions. Customer N1 comments the conduct of connection N2→N3 for a session of time Tobs.

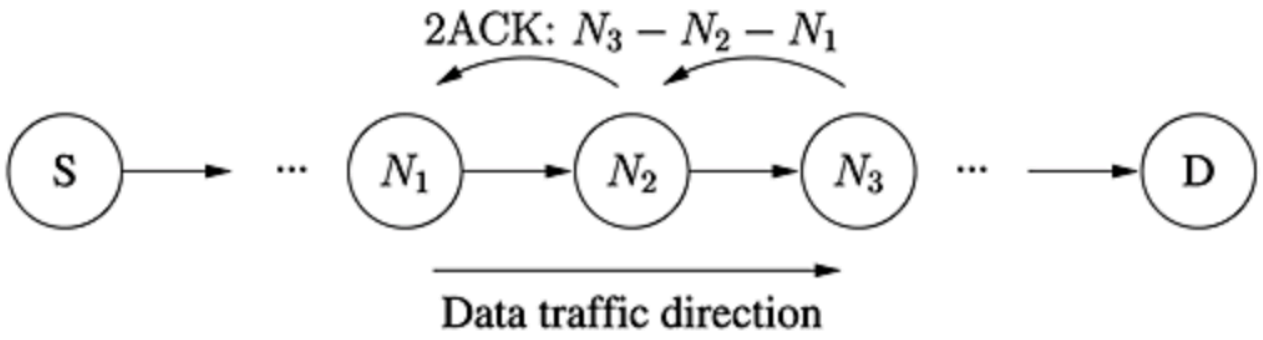


Fig 1.3: Settled way of two jumps in 2ack

At the remainder of the session, N1 decides the extent of losing 2ACK packages as Cmis/Cpkts and contrast it and a limit Rmis. In the event that the extent is more noteworthy than Rmis, connect N2→N3 is declared making trouble and that specific connection is being expelled from the directing table. Since just a partition of the get information package are recognized, Rmis could disentangle Rmis > 1-Rack disregard false cautions reason by such a mostly affirmation method. Each customer getting such a 2ACK bundle comments the connection N2→N3 as acting up and entireties it to the dark records rundown of such acting up joins that it controls. At the point when a customer starts its own information activity after, it will abstain from utilizing such acting up interfaces as a piece of its way.

2. Best effort fault tolerant routing

The plan objective of BFTR is to give bundle directing administration high conveyance proportion and low overhead in nearness of acting up hubs.BFTR works in a redundant network with no single point of failure (Fig 1.4). Rather than judging whether a way is great or awful, i.e., regardless of whether it contains any getting into mischief hub, BFTR assesses the steering plausibility of a way by its conclusion to-end execution (e.g. parcel conveyance proportion and deferral)[4]. By persistently watching the directing execution, BFTR powerfully courses bundles through the most doable way. BFTR gives a proficient and uniform answer for a wide scope of hub mischievous activities with not very many security suspicions. The BFTR calculation is assessed through both investigation and broad recreations. The outcomes demonstrate that BFTR significantly enhances the specially appointed directing execution within the sight of making trouble hubs.

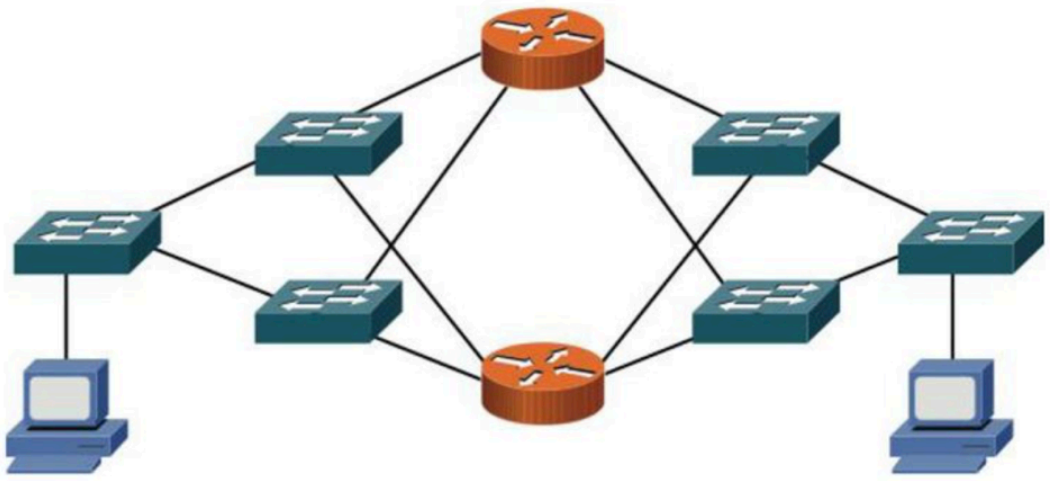


Fig 1.4: Redundant network with no single point of failure

3. Cooperative Bait Detection Scheme

The CBDS conspire contains three stages:

- The starting draw step.
- The invert following advance and
- The moved to responsive barrier step,

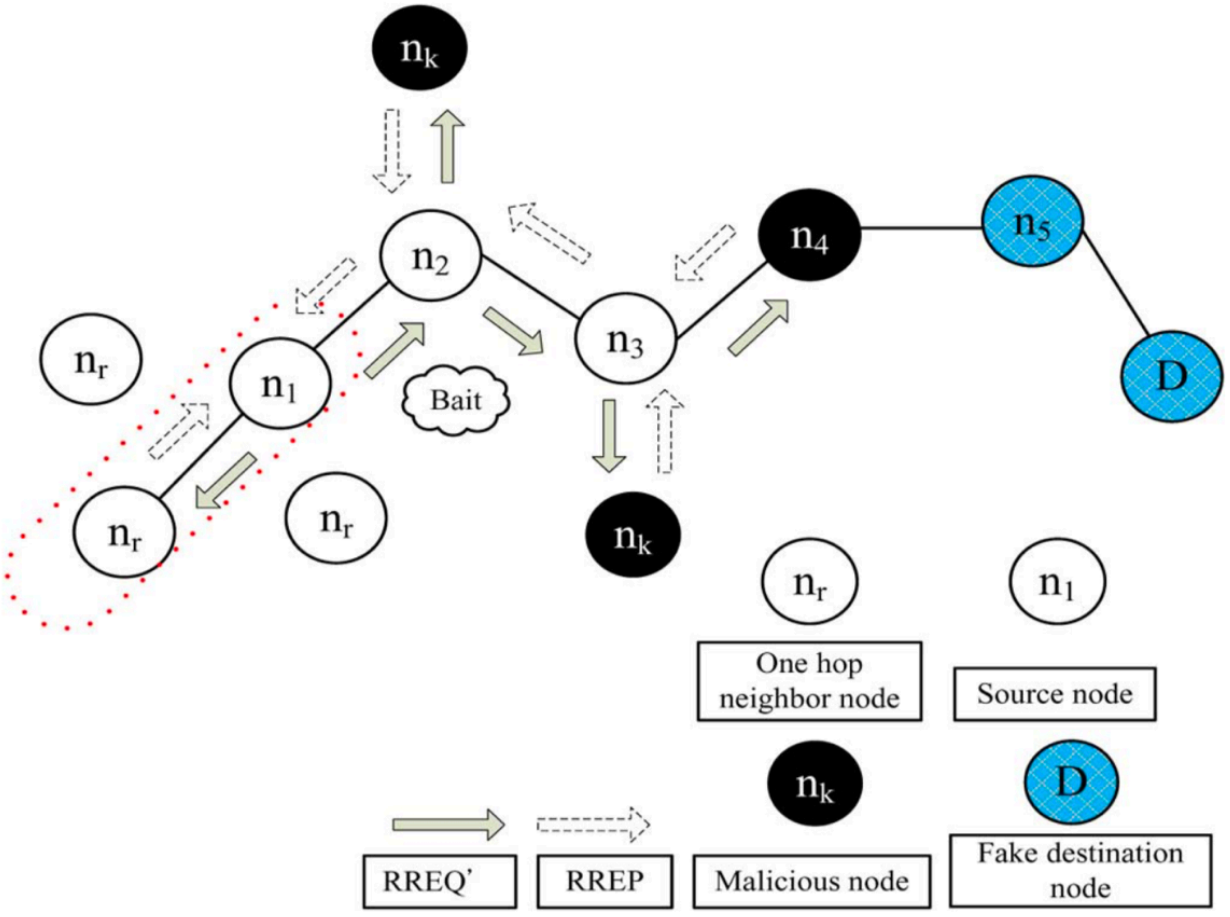


Fig 1.5: Random Selection of Cooperative baits

Initial Bait Step :

The objective of the goad mastermind is to allure a noxious center point to send an answer RREP by sending the catch RREQ that it has used to advance itself at the present time most briefest way to the center point that confines the packages that were changed over[1]. To accomplish this objective, the running with system is proposed to make the objective territory of the draw RREQ '. The source center

point consequently picks an adjacent center (Fig 1.5).

Reverse Tracing Step :
The inverse after advance is utilized to perceive the acts of threatening centers through the course answer to the RREQ' message. On the off chance that a toxic center point has gotten the RREQ' , it will answer with a false RREP. In like way, the switch following undertaking will be composed for center tolerating the RREP, with the objective to determine the questionable information and the out of the blue confided in zone in the course.

Reactive Defense Step :
After the above starting proactive shield (stages 1 and 2), the DSR course revelation process is started. Exactly when the course is set up and if at the objective it is found that the package conveyance proportion in a general sense tumbles as far as possible, the location design would be initiated again to perceive for consistent help and constant response profitability The edge is a varying a motivation in the range that can be adjusted by the present framework viability.

1.7 Problem Statement

This project aims at detecting the blackhole nodes in the network using modified bait scheme and prevent them by blacklisting them and create alternate path from source to destination that doesn't include any blackhole node.

1.8 Proposed Approach

The problem with MANETs is that it is susceptible to black hole attacks. Black hole nodes are the attacker nodes that pretend as if they have route to destination and drop all the packets received from the source. Hence there's data loss happening in the network. In the proposed algorithm that is modified bait scheme , we set the destination to be an invalid node and broadcast the bait request to all nodes in the network and only the blackhole nodes respond to this request saying they have route to destinations. Once we have identified them, we blacklist them and create an alternate path between source and destination which doesn't include any of the blackhole nodes. Thus data gets delivered from source to destination safely without getting lost.

1.9 Summary

This section gives the introduction to the concepts of networking in MANETs and other terms related to the topic.Also it gives information about MANETs and black-hole attacks and the existing methodologies that are being used to detect and prevent black-holes in MANETs.

Chapter 2

SOFTWARE REQUIREMENTS SPECIFICATIONS

2.1 Software Requirement Specification

Requirement specification is the activity of translating the information gathered during analysis into a requirements document. A Software Requirement System is an extensive description of the intended reason and environment for programming being worked on. The SRS completely portrays what the product will do and how it will be relied upon to perform.
The SRS report enrolls every single vital necessity that are required for the improvement. To determine the prerequisites, we need clear and careful comprehension of the items to be created. This is set up after itemized communications with the venture group and customer. Transportability, viability, impression, security and speed of recuperation from adverse occasions are assessed.

2.2 Operating Environment

In this project we have modeled an approach for dynamic resource allocation in wireless environment and to evaluate the stability of the system under consideration. The users can focus on the specific system level design issues, without having to concern about low level design and infrastructure. Linux as operating system which is open source. The hardware and software requirements are as mentioned below:

Hardware Requirements

- Processor: Any processor with speed above 500 MHz.
- RAM: 512Mb(minimum).
- Input Devices: Standard Keyboard and Mouse.
- Output Devices: High Resolution Monitor.

Software Requirements

- Operating system: Linux
- Programming tool: NS 2.34
- Documentation: overleaf.com

2.3 Functional Requirements

Functional requirements are a formal way of expressing the expected services of a project. We have identified the functional requirements for our project as follows:

- Ability to work with the hundreds of nodes with efficient resources utilization. If network contains more number of nodes then also each node must use the energy efficiently to sending and receiving the information from its neighboring nodes without failure.
- Ability to use the energy based on end-to-end delay among the nodes. For increasing the life time of network each node must use their energy to increase the life time of network.

2.4 Non-Functional Requirements

Non functional requirements are the various capabilities offered by the system. These have nothing to do with the expected results, but focus on how well the results are achieved.

- **Consistency**:The system must be consistent. It should not have any erratic behavior during operation.
- **Error Prevention and Correction**:The system should be able to capable of handling large number of nodes. If any errors occurred it must show the error with statement and that statement line number.
- **Maintainability**:The project work requires optimum scores in maintenance issues. All The algorithms being properly implemented with all consideration of user deployment, the application don't have any requirement of maintenance.

- **Extensibility**: The project work is open for future modification.

2.5 Applications

The applications for the project are:

- This system can be used for military operations.
- This system can be used secure and fast communication during law enforcement operations.

2.6 Advantages

This system has the following advantages:

- Reduces the network overhead and vulnerabilities by applying bait procedure from each and every node that is within its radio range.
- High Throughput and Less Delay in MANETUsing Bait Procedure.

2.7 Summary

In this chapter, we saw a brief description of SRS, followed by Hardware requirements and then software requirements along with functional and non-functional requirements. It also states applications and advantages of our project.

Chapter 3

HIGH LEVEL DESIGN

3.1 High Level Design

High level design is concerned with distinguishing programming components which determine relationships among components, along these lines indicating programming structure and giving blue print to the archive stage. It is essential for ETL engineers to comprehend the framework stream with functions.

Design is a phase which when executed outcomes a reasonable definition of how an issue best case scenario tackled. It depicts the structure of the product to be executed, the information which is a piece of the framework, the interfaces between the framework components and now and again the calculation utilized. Designers don't touch base at a completed design promptly yet build up the design iteratively h=through various diverse versions. The design procedure includes adding custom and detail as the design is created with constant backtracking to remedy prior designs.

3.2 Design Considerations

The designing procedure of a product includes consideration of numerous perspectives. Every one of these viewpoints ought to mirror the objectives the product is endeavoring to accomplish. A portion of the viewpoints like similarity, dependability, practicality, convenience, security and so on ought to be satisfied by each design. **Goals and Constraints**:

- Protecting sensitive data from malicious users.
- To provide confidence to the users that their data is safe.
- Detection and Prevention of black-hole attacks in MANETS.
- The overall simulation is shown using NS2.

3.3 System Architecture

The purpose of Architecture diagram outlines the set of signification decisions about the software system including the selection of the structural elements and interfaces of which the system is composed.

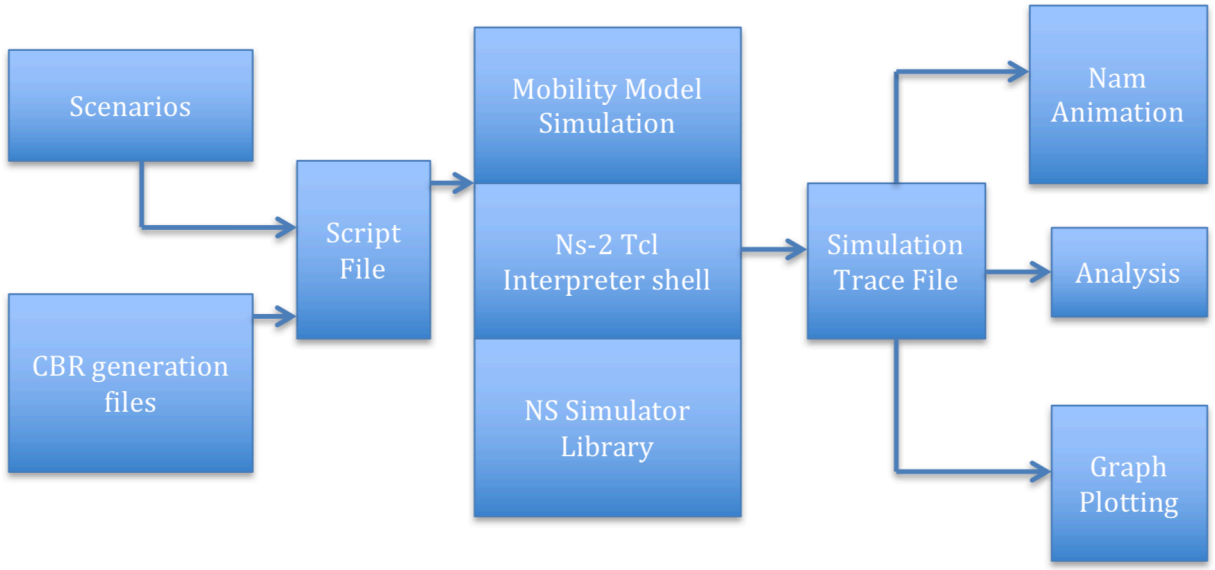


Fig 3.1: System Architecture

At the simulation layer, NS uses OTcl (Object Oriented Tool Command Language) programming language for interpreting the user simulation scripts. OTcl is an object oriented extension of Tcl.
While OTcl content is being deciphered, NS at the same time makes two principle examination reports as records. NAM (Network Animator) question is made to demonstrate the visual animation of the simulation and a Trace protest demonstrates the conduct of the considerable number of articles in the simulation. Previous is ".nam" document utilized by NAM programming that joins NS. Later is a ".tr" record that incorporates all simulation follows in the content arrangement. The ".tcl" documents in the content manager compose and break down the aftereffects of the ".tr" record utilizing "feline", "awk", "wc" and "grep" summons of Unix Operating System.
In view of the qualities accomplished in simulation, a diagram is plotted. xgraph is utilized and the order gnuplot is utilized to plot the chart. In the wake of plotting the chart, a comparison is made between the proposed demonstrate and the current model.

3.4 Sequence Diagram

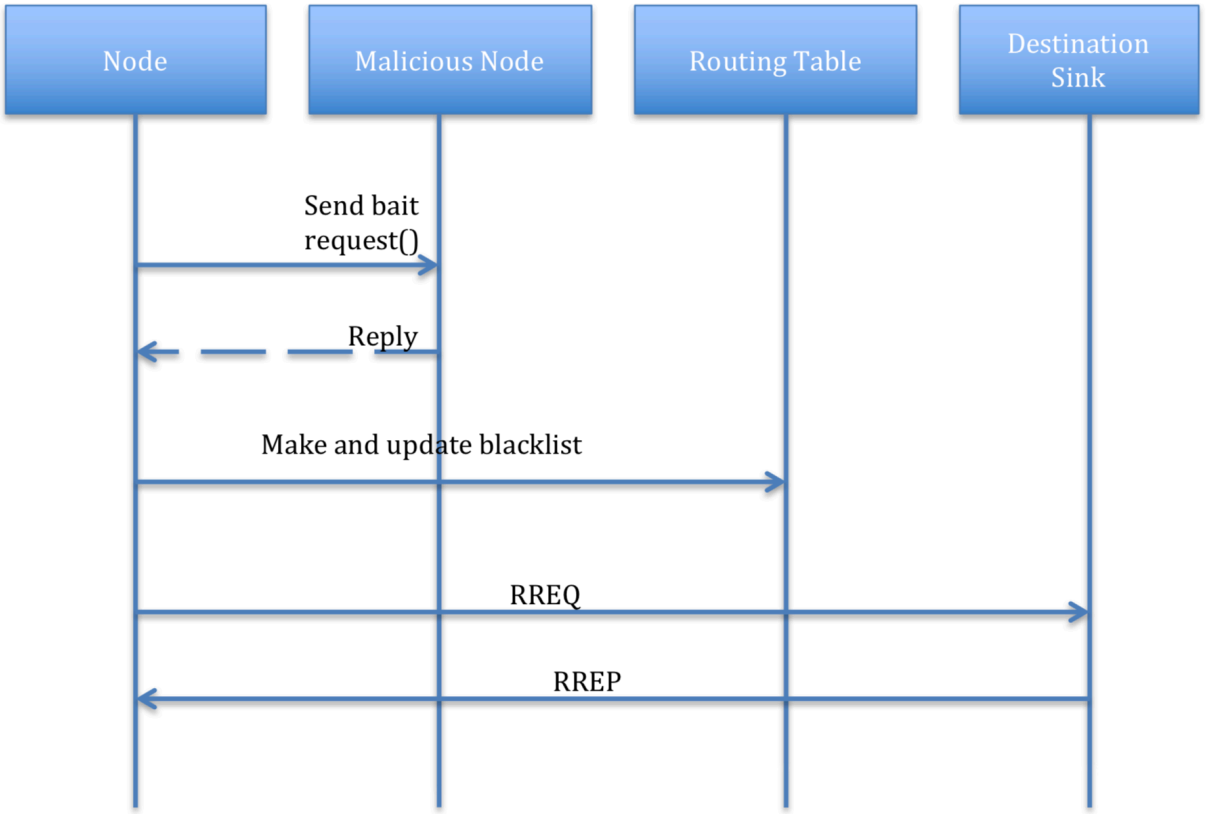


Fig 3.2: Sequence Diagram

A succession chart or sequence model in unified modeling language (UML) is a sort of interaction graph that demonstrates every component as protest, how they work with each other and in what arrange. Arrangement Diagram has parallel vertical lines speaking to life savers of each question. It demonstrates every one of the messages being traded among those articles. Arrangement graph speaks to the activation time of each protest.

The network has 60 nodes in total. The multiples of 4 and 5 are configured to be malicious. The blacklist is declared as an array of length 20. This array length can be increased or decreased accordingly.

3.5 Flow Diagram

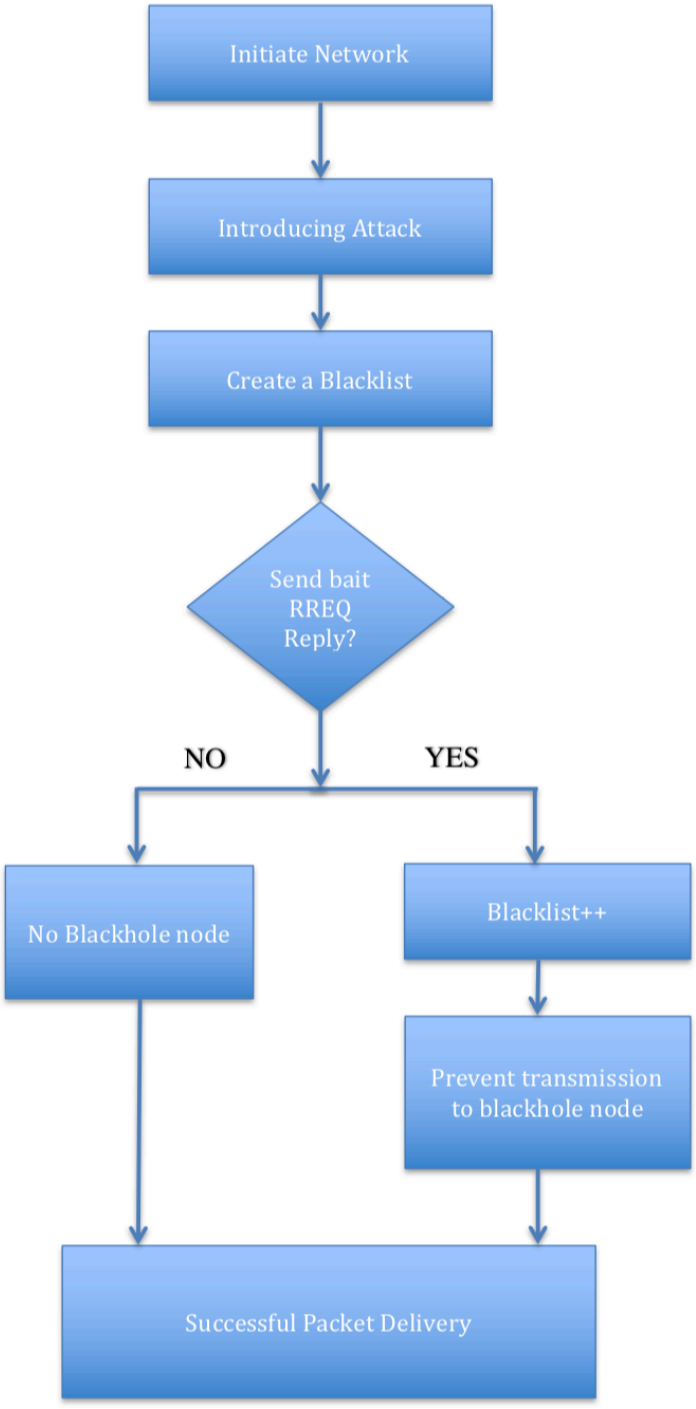


Fig 3.3: Flow Chart of the proposed system

The following are mentioned in the Fig 3.3:
1. Initiate Network :- A MANET network is created using Network Simulator ns2 version 2.34. The network consists of mobile and randomly distributed nodes.
2. Introducing Attack :- We introduce a blackhole attack by configuring few of the nodes as malicious in the network.
3. Create a Blacklist :- Assign an array of some specified length to which we add the index of the blackhole nodes present in the network.
4. Send bait RREQ :- The source node send a bait request RREQ to the adjacent node with a destination node index which is not present in the network.
5. Receive a reply RREP :- If a reply is received, the index of the node which sends a reply to this previously send RREQ, the index of that node is added to the blacklist to ensure that no packets are sent to this malicious node.
6. No reply received :- If no reply is received, this means there are no blackhole nodes in the network and hence, the packet delivery takes place normally.
7. Prevent transmission to blackhole nodes :- Each time a source node wants to a send a packet to a destination, it avoids sending that packet to a blackhole node by referring the blacklist.
8. Successful Packet delivery :- After the detection and prevention of blackhole nodes, the packet is delivered successfully.

3.6 About NS2

ns is a object specific simulator, written in C++ with an OTcl translator as a frontend. In NS-2, the frontend of the program is composed in TCL. The backend of NS-2 simulator is composed in C++. On compilation of TCL program, a follow record and nam document are made demonstrating the development example of hubs and monitoring the quantity of parcels sent, connection type,etc at each case of time. The network parameters can be expressly mentioned amid the creation of situation and connection design records utilizing the library functions of the simulator.

3.7 Summary

The system architecture is shown in section 3.2 and data flow diagram is shown in section 3.4. The functionality of each component is explained in brief.

Chapter 4

DETAILED DESIGN

4.1 Purpose

This chapter gives a detail design of the project. It concentrates on the different modules that have been developed. The basic purpose of this project is to detect and prevent the Black-hole attack in the self-configuring nature of network.

4.2 Processing Steps

To prevent the black-hole attack in the network we use the Modified Bait Scheme.
The steps involved in the processing are:

Step 1: A MANET network is created using Network Simulator ns2 version 2.34. The created network consists of mobile and randomly distributed nodes and each node in the network acts as a host as well as router.

Step 2: We introduce a black-hole attack by configuring few of the nodes as malicious in the network. These malicious nodes drops the routing packets but does not forward packets to its neighbours.

Step 3: Assign an array of some specified length to which we add the index of the black-hole nodes present in the network.

Step 4: The source node send a bait request RREQ to the adjacent node with a destination node index which is not present in the network. Bait phase is initiated when bait RREQ' is used for initial routing and waiting for the reply.

step 5: When the RREQ' received by a malicious node, then it will reply with a forged false RREP. After that by using that false reply we can find out the malicious node.

Step 6: If no reply is received, this means there are no black-hole nodes in the network and hence, the packet delivery takes place normally.

Step 7: After the detection and prevention of black-hole nodes, the packet is delivered successfully.

4.3 Modified Bait Scheme

1.Make a blacklist array
2.Send Bait request
3.Check for nodes that give RREQ.
4.Note the index values of these nodes and add them to the blacklist.
5.Send the blacklist to the routing table.
6.Prevent transmission of the data from or to these nodes.

4.4 Summary

This Chapter gives detailed design of the project. It explains the purpose and processing steps and also brief explanation of the module.

Chapter 5

IMPLEMENTATION

5.1 Implementation

Implementation is an important phase in the development life cycle. This phase is the translation of the system requirements and specifications into a working model to fulfill the real time services. Key functionalities identified from the design stage are converted into functions that are executable using appropriate programming languages.
Thus implementation phase is always preceded by important decisions relating to the language selection, platform selection etc. These decisions are influenced by various factors such as response time required, security concerns, and data management concerns among many others. These decisions also affect how well the final product functions.

Implementation steps:
1. The network is setup to run the default AODV protocol
2. The Black-hole node is introduced into the network which results in packet drop.
3. Black-hole detection step is employed.
4. Prevention methodology is applied which results in successful data delivery

5.2 Programming Language Selection

While implementing a software system, programming language selection is one of the key decisions that has to be made. Since there are so many programming languages to choose from, it is easy to get lost in the intricacies of each. The choice of programming language however depends on a variety of factors. Some of the factors that should be kept in mind while selecting the programming language are:

- Programming language skill of each team member working on the project.
- Platform support and portability.
- Applicability to the problem domain.
- Availability of the necessary libraries to setup the environment.

By considering all the above factors, we have chosen ns2 for simulation. Ns2 had the libraries that

was required for the creation of the networking environment. Moreover, ns2 was an application which everyone in the team was comfortable with. Ns2 version 2.34 is used for better support and lesser bugs.

5.3 Platform Selection

We have used linux platform for our project.The following are the reasons:

1.Flexibility
This stems from the desktop at the same time, since Linux is such an incredibly adaptable working framework, it's wrong to confine adaptability to the desktop alone. Here's the thing: With Linux, there is constantly in excess of one approach to deal with an errand. Add to that the capacity to get extremely imaginative with your critical thinking, and you have the makings of a far unrivaled framework. Windows is about as rigid as a working framework can be.

2.Shell Scripting
Shell Scripting is better because of the following reasons:

- One can write a script to initialize something at boot time of the system, so that it doesn't have to be done manually.
- Its easy to kill or start multiple applications together.
- Better automation

3.Command Line
This is another thing where I shouldn't need to state significantly more than the title. The Linux order line can do about anything you have to work in the Linux working framework. Truly, you require a touch of information to do this, however similar remains constant for the Windows command line.
The greatest distinction is the sum you can do when met with only the charge line. On the off chance that you had to administer two machines through the order line only , you would rapidly see exactly how better the Linux CLI is than the incomprehensibly underpowered Windows CLI.

5.4 Implementation Steps

1.SETUP THE NETWORK

The network is setup so that normal AODV routing takes place. An Ad Hoc On-Demand Distance Vector (Fig 5.1) is a coordinating convention intended for remote and adaptable extemporaneous frameworks. This convention develops courses to objectives on demand and support both unicast and multicast directing. The AODV convention was commonly made by Nokia Research Center, the University of California, Santa Barbara and the University of Cincinnati in 1991.

The AODV convention fabricates courses between centers just if they are requested by source centers. AODV is consequently seen as an on-ask for calculation and does not make any additional action for correspondence along joins. The routes are kept up as long as they are required by the sources. They are self-start and circle free other than scaling to different versatile centers.

In AODV, frameworks are quiet until the point that the moment that associations are developed. Framework center points that need associations impart a demand for association. Whatever is left of the AODV centers forward the message and record the center point that requested an association. Along these lines, they make a progression of brief courses back to the requesting center point.

A hub that gets such messages and holds a route to a coveted hub sends a retrogressive message through impermanent courses to the asking for hub. The hub that started the demand utilizes the course containing minimal number of jumps through different hubs. The passages that are not utilized as a part of directing tables are reused after some time. On the off chance that a connection comes up short, the directing blunder is passed back to the transmitting hub and the procedure is rehashed.

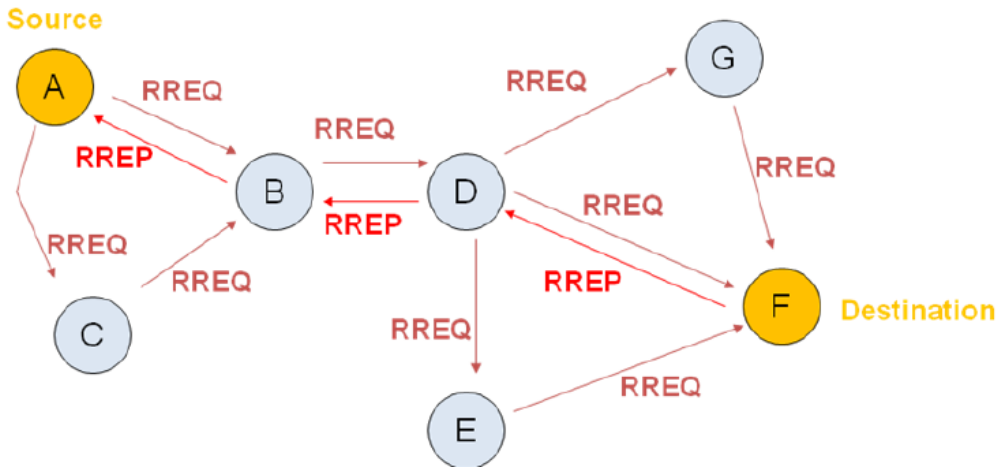


Fig 5.1: Simple Aodv

2.INTRODUCE BALCK-HOLE NODE

Black-hole nodes are introduced into the system. Black hole attack is one of the security threat in which the traffic is redirected to such a node that actually does not exist in the network so that the data packets are dropped instead of being received at the destination (Fig 5.2)[1]. It's a real world analogy to the black hole in the universe in which things disappear. The node presents itself in such a way to the node that it can attack other nodes and networks knowing that it has the shortest path. Figure given below shows black-hole attack:

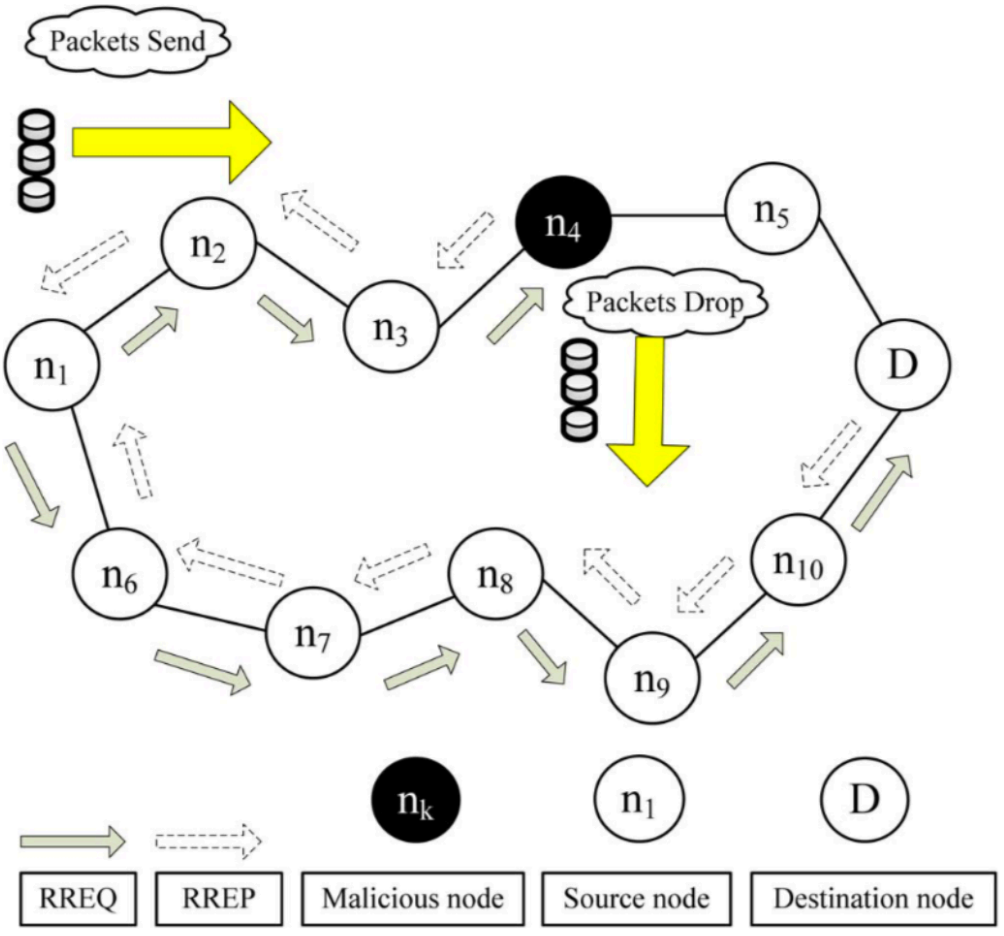


Fig 5.2: Data packet drop by Blackhole node n4

3.DETECTION OF THE BLACK-HOLE NODE

A bait Route Request (RREQ) is sent. The bait RREQ has the destination of a node which is not present in the network. Usually, in normal circumstances a RREQ receives a Route Reply (RREP) only if that particular node is present in the network and the node which sends the RREP should have the least distance from the source to destination node. But here if it gets a RREP then we can confirm that the node which send the RREP is a malicious node.

4.PREVENTION OF BLACK-HOLE ATTACK

A list is created which consists of all black-hole nodes.

```
for(int i=0;i<n;i++)
{
    if( blackholenodes[i] == index )
    {
        return
    }
}
```

Fig 5.3: Creating the blacklist

All the nodes that give RREP for the bait RREQ is added into the blacklist.

```
if( timesrouted == 1)
{
    if( foundb < 20 )
    {
        blackholenodes[foundb]=rp->rp_dst;
        foundb++;
    }
}
```

Fig 5.4: Addition of blackhole nodes to the blacklist

After this step, the blacklisted nodes are sent to the routing table in order to stop transmission to or from these nodes.

5.5 Coding Conventions

The developed program may not be maintained by the developer throughout its lifetime. Thus it is highly important to make the code comprehensible to any new readers as well. Making the code more readable helps in changing and maintaining the code even in the future.Comments are given for all important and basic changes.

- **Naming variables and constants**: The naming is done such that the name indicates the variable's/ constant's obvious meaning without too much ambiguity.
- **Comments**: Appropriate comments are provided in the code to make it easier for another person to review the code with much ease.

5.6 Summary

This chapter dealt with the various techniques used in the development of the project,starting with the language and platform selection to finally explain the entire process of implementation steps.

Chapter 6

SIMULATION

6.1 Simulation

Simulation is the imitation of the operation of a genuine procedure or framework. The demonstration of reenacting something initially requires that a model be produced; this model speaks to the key qualities, practices and functions of the chose physical or theoretical framework or process. The model speaks to the framework itself, while the simulation speaks to the operation of the framework after some time.
Simulation is utilized as a part of numerous contexts, for example, simulation of innovation for execution optimization, wellbeing building, testing, preparing, education, and computer games. Regularly, PC tests are utilized to examine simulation models. Simulation is likewise utilized with logical modeling of normal frameworks or human frameworks to pick up knowledge into their functioning, as in economics. Simulation can be utilized to demonstrate the inevitable genuine impacts of elective conditions and blueprints. Simulation is likewise utilized when the genuine framework can't be locked in, in light of the fact that it may not be available, or it might be risky or unsuitable to connect with, or it is being designed yet not yet constructed, or it might essentially not exist

6.2 NS2 Simulation

NS2 stands for Network Simulator Version 2. It is an open-source occasion driven simulator designed particularly for examine in PC communication networks.

- Ns is a discrete event simulator targeted at networking research.
- It is primarily Unix based.
- NS2 provides support to simulate bunch of protocols like TCP,UDP and DSR.
- NS2 Uses TCL as its scripting language.

6.3 Simulation Testing

In order to check how the transmission takes place under different aodv routing files we do simulation.
Simulation Test 1:

Simulation Test Case ID	Test case 1
Description	To test the default aodv routing protocol in presence of blackhole nodes.
Input	Link aodv.cc file to point to default aodv.
Expected Output	No transmission takes place.
Actual Output	No transmission took place.
Remarks	Passed

Table 6.1: Simulation Test Case 1

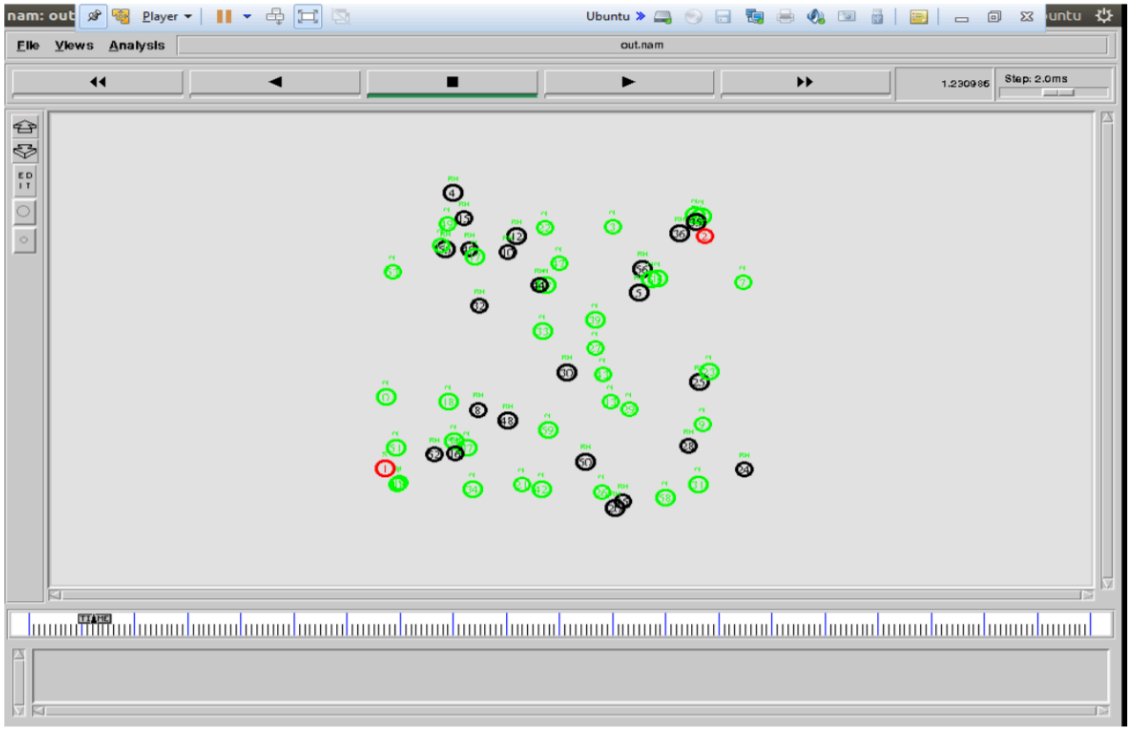


Fig 6.1: Simulation Diagram for Test Case 1

Simulation Test 2:

Simulation Test Case ID	Test Case 2
Description	To test the aodv routing with the modified bait algorithm.
Input	Link aodv.cc file to point to blackhole_aodv which has modified bait algorithm.
Expected Output	Nodes should find an alternate path to the destination and packets must be sent from source to the destination.
Actual Output	Nodes found out an alternate path from source to destination and most of the packets from source were received at destination.
Remarks	Passed

Table 6.2: Simulation Test Case 2

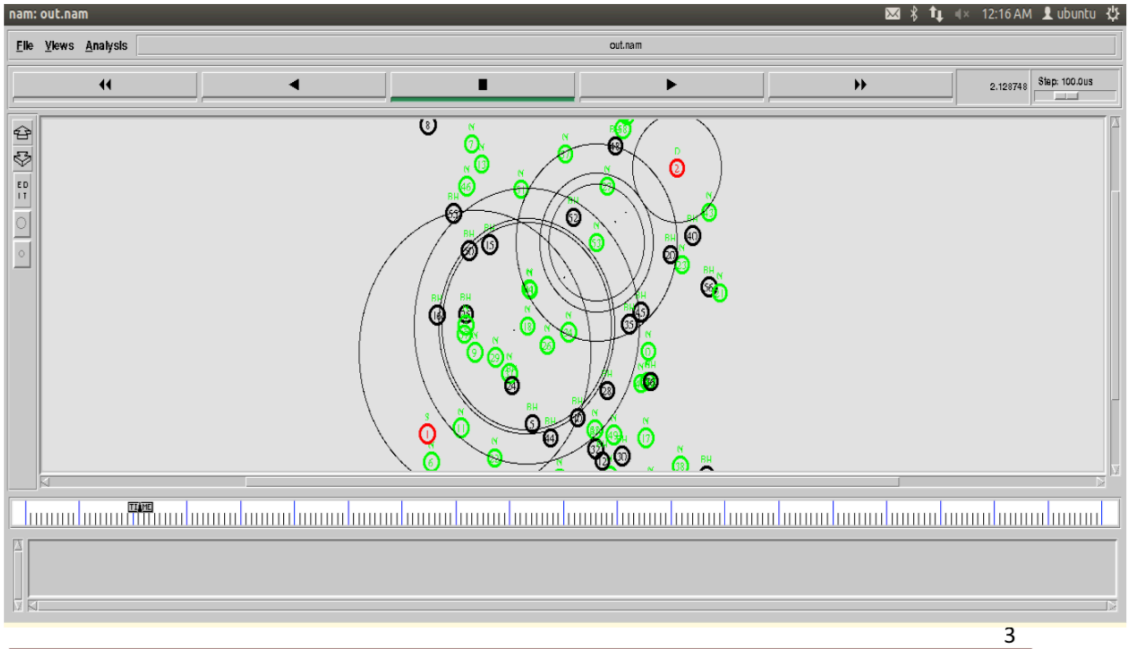


Fig 6.2: Simulation Diagram for Test Case 2

6.4 Summary

In this simulation, initially we set up a Mobile Ad Hoc network with 60 nodes.
In the first simulation test we link the ”defaultaodv” file to aodv.cc file. Once the link is established we run the tcl script followed by simulation which will show that no transfer of packets takes place between source and destination in presence of black-hole nodes in the network.
In the next simulation test, we link the black-hole AODV file which includes modified-bait scheme to aodv.cc. Once the link is established we will run the tcl script followed by simulation that shows that the nodes have chosen an alternate path from source to the destination that doesn’t include any of the black-hole nodes in the network. Hence the expected outcome matches with the actual outcome and the simulation test is passed.

Chapter 7

SIMULATION RESULTS

7.1 Simulation Results

As we mentioned earlier in simulation that,Initially we simulate the ”default aodv”(Simple Aodv),Fig 7.1, routing protocol and the agenda is no packets should get delivered from source to the destination. This happens because there cannot be any path establishment because of presence of black-hole nodes in the network. Initial Simulation is as follows,

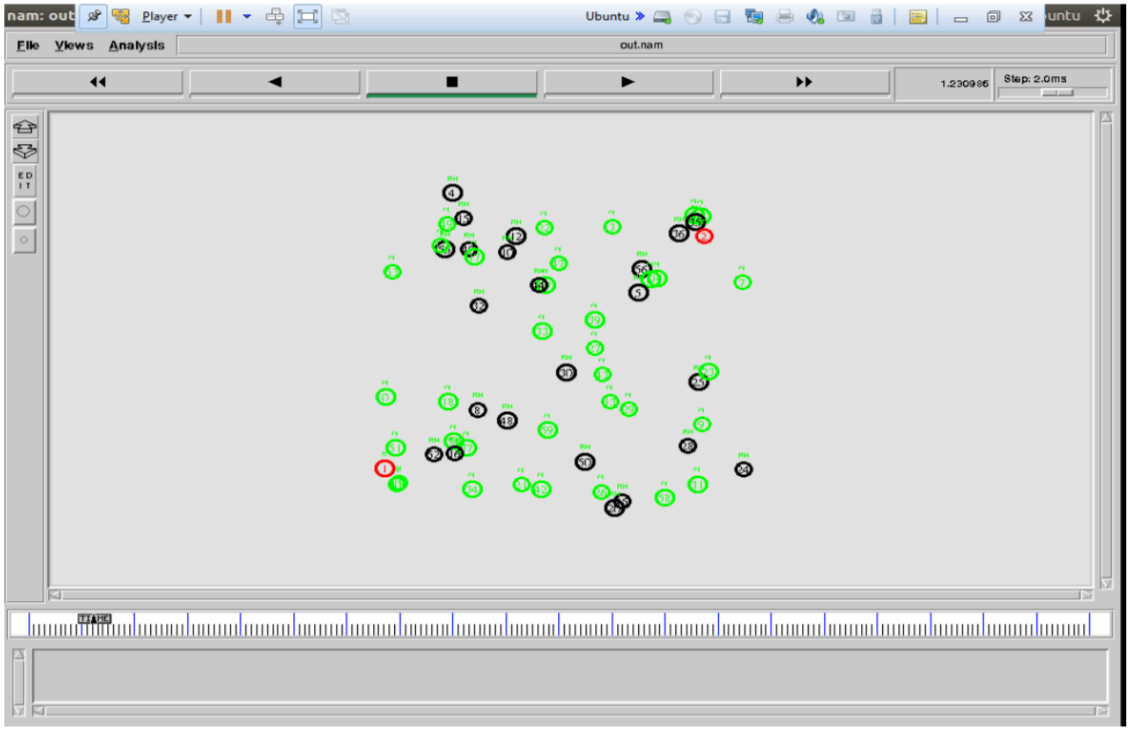


Fig 7.1: Simple AODV transmission with blackhole

In order to measure the performance we need to find something known as PDR(packet delivery ratio).PDR is the ratio of the no.of data packets received to the no.of data packets sent. Since ,in default aodv there is no transmission happening between source and destination, the packet delivery ratio must be zero.

In order to evaluate PDR, we write an awk script that analyzes the trace file of the simulation and generates the PDR value. The output of the awk file is as follows,

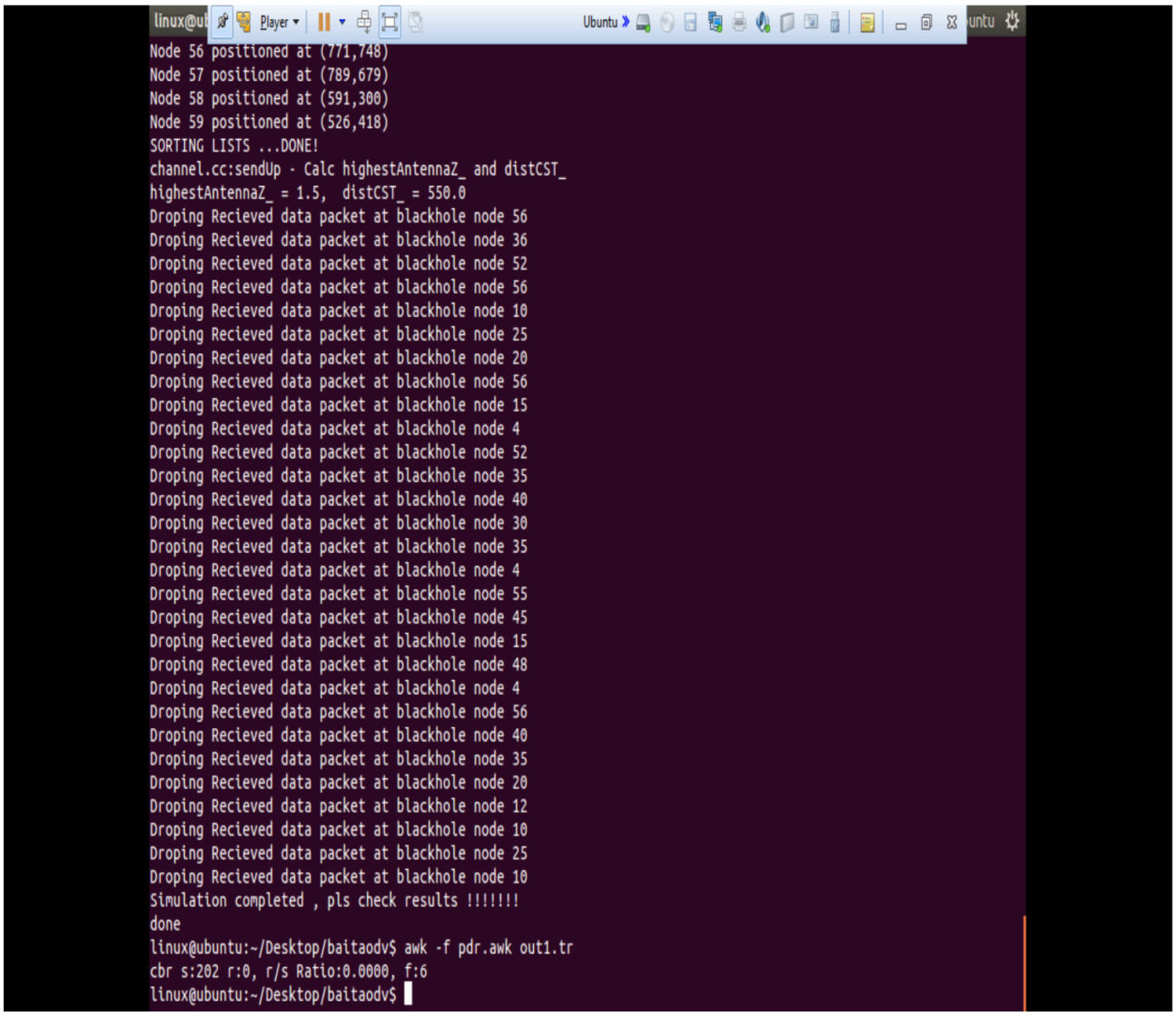


Fig 7.2: Awk Script showing PDR for simple transmission

In the above figure (Fig 7.2) it clearly shows that PDR is 0. Since there are blackholes nodes present in the network the default aodv doesn't generate any path between source and destrination and hence ratio of received to sent is 0.

Next, we need to analyze the throughput and delay that is present in the network, this is done by plotting the graphs with the values extracted from tracefile.

As we mentioned earlier in simulation that ,In the next simulation test we simulate the ”blackhole aodv” (Fig 7.3) routing protocol and the agenda is nodes should find an alternate path from source to the destination without including any of the black hole nodes in the network. And all data packets should get delivered from source to the destination in the newly established path. This happens because we are blacklisting all the nodes that respond to the bait request and while establishing the path we ensure none of the nodes in the blacklist are included.The simulation is as follows:

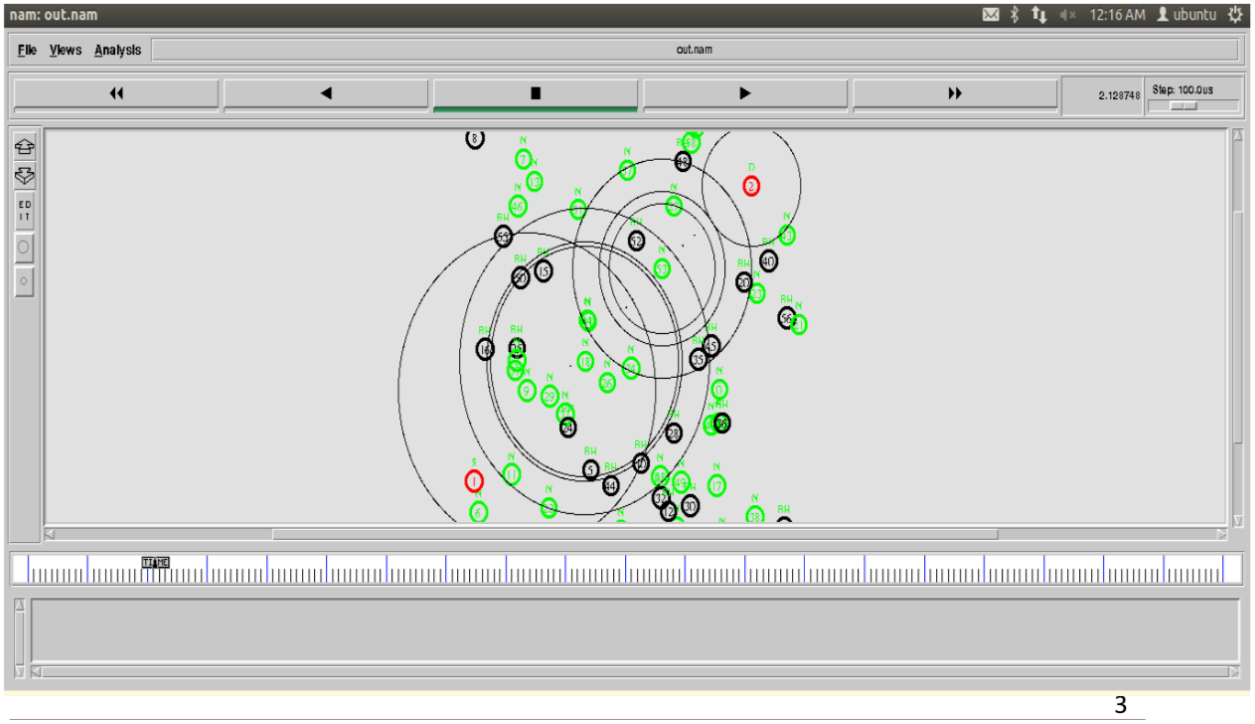


Fig 7.3: Transmission after applying Modified Bait

We can see from the image that the nodes have found out an alternate path from source to destination without including any of the blackholes. Since alternate path is established all the data packets sent from source must be received at the destination. Hence the PDR is nearly 100 percent.

In order to evaluate PDR, we write an awk script that analyzes the trace file of the simulation and generates the PDR value. The output of the awk file is as follows,

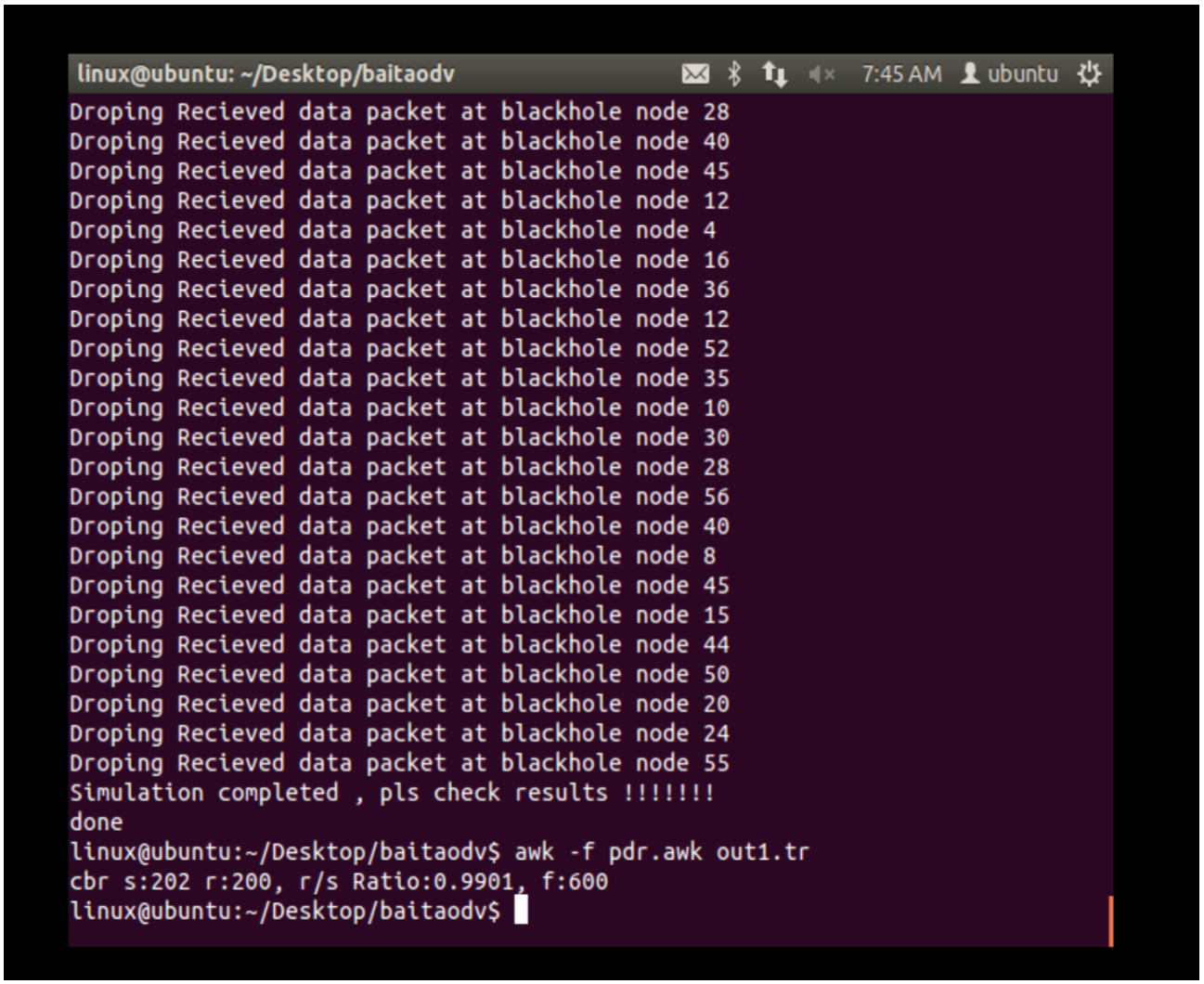


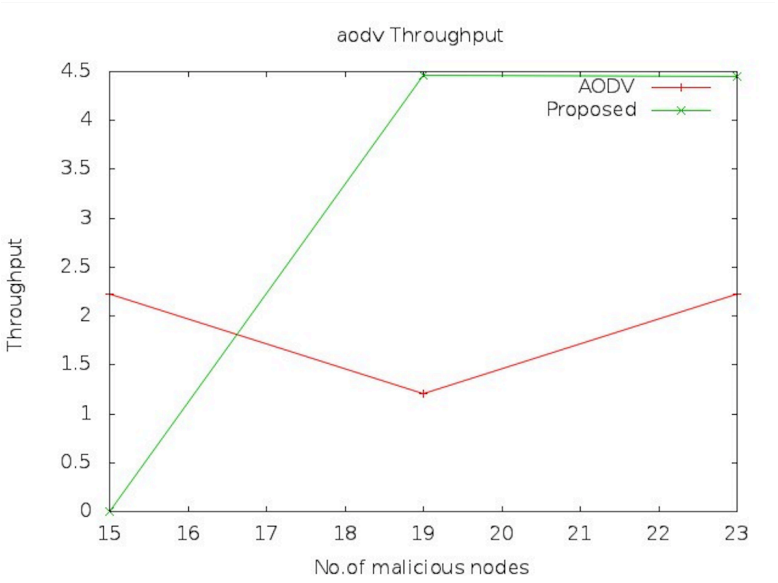
Fig 7.4: Awk Script showing PDR after applying Modified Bait Scheme

We can see from Fig 7.4 ,that almost all the packets sent from source were clearly received at the destination even in the presence of blackholes. This happens because of the fact,that we are blacklisting the blackhole nodes and are not considering for route creation. The value of PDR from above figure is 99.01 percent that is most of the packets were received at the destination.

Next, we need to analyze the throughput and delay that is present in the network, this is done by plotting the graphs with the values extracted from tracefile.

THROUGHPUT

Throughput is the number of data packets that were sent successfully from source to the destination.

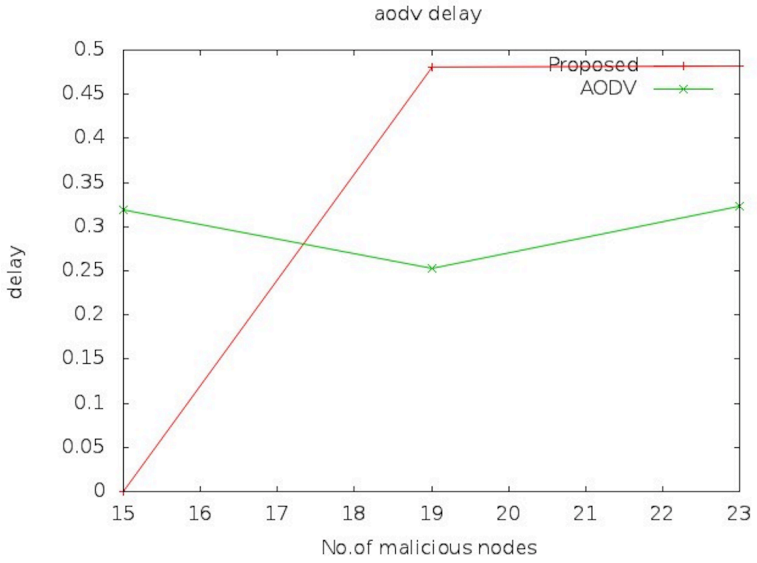


Graph 7.1: Graph plotted for throughput against number of malicious nodes.

In throughput graph, as we can see that throughput for the "default aodv" is almost constant as there is no transmission taking place between Source and destination. In case of modified bait scheme which is proposed approach, throughput increases with increase in number of malicious nodes thus depicting the effectiveness of proposed approach.

DELAY

Delay is nothing but the latency occurred in the round trip of data packets.Ideally there shouldn't be any delay in default aodv since there is no transmission of data packets.

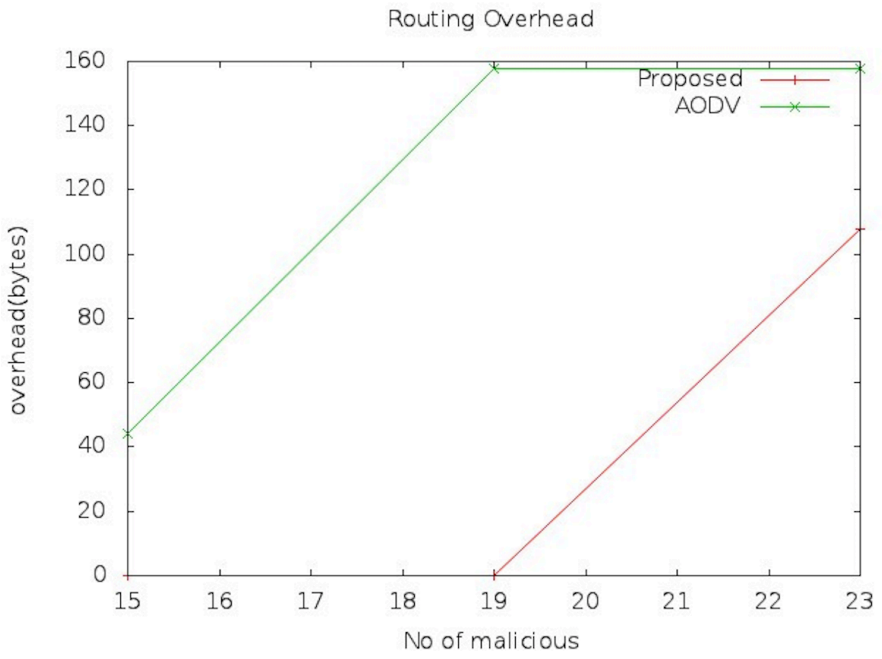


Graph 7.2: Graph plotted for delay against number of malicious nodes.

In delay graph, As we can see that delay for the "default aodv" routing protocol is almost constant as there is no transmission taking place between source and destination. In case of modified bait aodv (proposed) sceme, the delay increases as the number of malicious nodes increases. This happens because, when the number of malicious nodes increases it takes time to establish path between source and destination nodes.

ROUTING OVERHEAD

Routing overhead tell you how many data packets are needed for communation in a network to take place.It is the ratio of packets sent to packets received.



Graph 7.3: Graph plotted for routing overhead against number of malicious nodes.

In Routing over head, as we can see that the routing overhead for the “default aodv” is very high compared to modified bait scheme, this happens because of presence of malicious nodes in default aodv. It finds it very difficult to establish any path between source and destination, thus increasing routing overhead. And in modified bait scheme since we have blacklisted all the blackhole nodes, establishing a route between source and destination becomes simpler.

7.2 Summary

In this chapter, the experiments carried out and the results obtained from these experiments are analyzed. The Packet delivery ratios along with throughput and delay are computed.

Chapter 8

CONCLUSION

With the assistance of above graphss, it demonstrates that Modified lure plan will effective to identify and keep the blackhole assault on MANET under AODV protocol.
Proposed algo will give security to MANET topology and keep blackhole from the dynamic route. Proposed work will be light weight and will accomplish great throughput, parcel conveyance proportion with less delay of bundles. Indeed, even within the sight of blackhole hubs dependability of the network stayed as an ordinary network. From above outcomes, we conclude that the proposed plan will proficiently distinguish and avert blackhole assault under Mobile Adhoc Network(MANETs).

8.1 Assumption

We consider a adhoc wireless network with the set of nodes and assuming that there is a possibility of one or more attacker nodes. Thereby we implement our proposed algorithm in order to prevent and remove the attacker node from the network.

8.2 Limitations

We have used only one parameter for detecting the black-hole nodes and that parameter is nothing but the reply to the bait request. Whichever nodes reply to the bait request will be pushed into the black list. We may consider other parameters as well.

8.3 Future Enhancement

- This implementation can be further improvised by considering other parameters of the network for identifying and blacklisting malicious nodes.
- This approach solely concentrates on sending bait request and identifying malicious nodes and thereby blacklisting them.
- This method can be implemented for networks which are more stable for more efficient routing.
- This can be implemented for AMMNETs to avoid network partitioning.

Bibliography

1 Jian-Ming Chang, Po-Chun Tsou, Isaac Woungang, Han-Chieh Chao, and Chin-Feng Lai, *"Defending Against Collaborative Attacks by Malicious Nodes in MANETs: A Cooperative Bait Detection Approach,"* IEEE SYSTEMS JOURNAL, MARCH 2015.

2 Hesiri Weerasinghe, IEEE Student Member, Huirong Fu, IEEE Member "*Preventing Cooperative Black Hole Attacks in Mobile Ad Hoc Networks: Simulation Implementation and Evaluation"*.

3 Dhiraj Nitnaware, *Black Hole Attack Detection and Prevention Strategy in DYMO for MANET*.(2016)

4 Y. Xue and K. Nahrstedt, *"Providing fault-tolerant ad hoc routing service in adversarial environments,"* Wireless Pers.Commun., vol. 29, pp. 367– 388, 2004.

5 K. Liu, D. Pramod, K. Varshney, and K. Balakrishnan, *"An Acknowledgement based approach for the detection of routing misbehavior in MANETs,"* IEEE Trans. Mobile Comput., vol. 6, no. 5, pp. 536–550, May 2007.

6 A. Baadache and A. Belmehdi, *"Avoiding blackhole and cooperative blackhole attacks in wireless ad hoc networks,"* Intl. J. Comput. Sci. Inf. Security, vol. 7, no. 1, 2010.

7 W. Wang, B. Bhargava, and M. Linderman, *"Defending against collaborative packet drop attacks on MANETs,"* in Proc. 28th IEEE Int. Symp. Reliable Distrib. Syst., New Delhi, India, Sep. 2009.

8 *Detection of Single and Collaborative Black Hole Attack in MANET* by Sathish, Arumugam, S.Neelavathy Pari.

9 *Bait Request Algorithm to Mitigate Black Hole Attacks in Mobile Ad Hoc Networks* by Ayanwuyi T. Kolade, Megat F. Zuhairi, Hassan Dao, Sohail Khan 2016.

10 *Cooperative Bait Detection Scheme to prevent Collaborative Blackhole or Grayhole Attacks by Malicious Nodes in MANETs* by Pradeep R. Dumne1, Arati Manjaramkar2 2016.

11 *Black Hole Attack Detection and Prevention Strategy in DYMO for MANET* by Dhiraj Nitnaware and Anita Thakur 2106.

12 L. Himral, V. Vig and N. Chand, *"Preventing AODV Routing Protocol from Black Hole Attack"*, International Journal of Engineering Science and Technology (IJEST) Vol. 3, No. 5, 2011.

13 Z. Alishahi, J. Mirabedini and M. K. Rafsanjani, "*A new method for improving security in MANETs AODV Protocol"*, Management Science Letters 2 (2012) 2271–2280.

14 Tariq Siddiqui and Tanveer Farooqui, *"A Survey on Malicious Node Detection in MANET,"* International Journal of Advanced Research in Computer Science and Software Engineering, V olume 4, Issue 12, December 2014.

15 W. Kozma and L. Lazos, "*REAct: resource-efficient accountability for node misbehavior in ad hoc networks based on random audits,"* In Proceedings of the second ACM conference on Wireless network security, pp.103 -110, 2009.