

Impersonation based Sybil attack to disrupt the Highest Degree Clustering algorithm in Mobile Ad Hoc Networks

Project report submitted in fulfillment of the requirement for the degree of Bachelor of Technology

in

Computer Science and Engineering/Information Technology

By

Prem Prakash (123210)

Under the supervision of

Mr Amol Vasudeva

to



Department of Computer Science & Engineering and Information Technology

**Jaypee University of Information Technology Waknaghat,
Solan-173234, Himachal Pradesh**

CERTIFICATE

Candidate's Declaration

I hereby declare that the work presented in this report entitled “Impersonation based Sybil attack to disrupt the Highest Degree Clustering algorithm in Mobile Ad Hoc Networks” in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology,

Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from August 2015 to May 2016 under the supervision of **Mr Amol Vasudeva** (Assistant Professor, Computer Science And Engineering Department).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Prem Prakash(123210)

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Mr Amol Vasudeva

Assistant Professor

Department: Computer Science and Engineering.

Dated:

ACKNOWLEDGEMENT

The project entitled as Impersonation based Sybil attack on mobile Ad hoc Networks using highest degree clustering is done to better understand the impacts of Sybil attack on system and networks. I would like to use this opportunity to express my gratitude to everyone who supported me throughout the course of this B.Tech project. I am thankful for their aspiring guidance, invaluable constructive criticism and friendly advice during the project work. I am sincerely grateful to them for sharing their truthful and inspiring views on a number of issues related to the project.

I am especially grateful to **Mr. Amol Vasudeva**, Project Supervisor, for his valuable suggestions, support and constant encouragement during the course of the project. His perpetual energy, motivation, enthusiasm and immense knowledge inspired me to discipline myself in efficiently executing my multiple responsibilities simultaneously. He has given us the push that we need to go further and to do more. So we take this opportunity to express our sincere gratitude and thanks to him for lending his cooperation.

TABLE OF CONTENT

Chapter No.	Title	Page No
1	Introduction	1
	1.1 Introduction	1
	1.2 Problem statement	11
	1.3 Objectives	11
	1.4 Methodology	12
	1.5 Organization	13
2	Literature Review	14
3	System Development	
	3.1 Algorithm for Sybil attack in highest degree clustering	28
	3.2 Problems with existing system	30
	3.3 Software Development model	31
	3.4 Diagrams	32
	3.5 Hardware requirements	34
	3.6 Software requirements	34
4	Performance Analysis	
	4.1 Complexity of algorithm	35
	4.2 Output at various stages	36
	4.3 Results	47
5	Conclusion	
	5.1 Conclusion	51
	5.2 Future Scope	52
	5.3 References	53
	APPENDICES	54

LIST OF ABBREVIATIONS

MANET ----- Mobile Ad hoc Networks

ID -----Identity

DSDV -----Destination Sequenced Distance Vector Routing

AODV ----- Ad hoc On-Demand Distance Vector Routing

MN -----Mobility Node

RWP -----Random Way Point Model

CH -----Cluster Head

GN. -----Gateway Node

CN -----Cluster Node

RIP -----Routing Information Protocol

JVM -----JAVA Virtual Machine

IDE -----Integrated Development Environment

JRE -----JAVA Run Time Environment

JDK -----Java Development Kit

GHz -----Giga Hertz

MB -----Mega Bytes

DFD -----Data Flow Diagram

LIST OF FIGURES

S. No	Title	Page No.
1	Sybil Nodes	2
2	Direct Communication	3
3	Indirect Communication	4
4	Fabricated Identity	5
5	Stolen Identity	5
6	Sybil attack in resource allocation	6
7	Sybil attack on distributed storage	7
8	Sybil attack effect on routing	8
9	Highest degree clustering algorithm	9
10	SDLC Life Cycle	12
11	Nodes of a MANET	19
12	Type of routing protocols	21
13	Cluster Structure in MANET	22
14	Cluster Head Selection Algorithm	22
15	Node-type in MANET	24
16	Nodes in MANET	25
17	Prototype Model	31
18	Zero level DFD	32
19	First level DFD	32
20	Second level DFD	33
21	Use case Diagram	33

22	Home page of application	37
23	Input page	38
24	Output with all legitimate nodes	39
25	current position of all nodes	40
26	Adjacency Matrix	41
27	Clustering	42
28	Message Details	43
29	Battery Levels	44
30	Sybil attack	45
31	Clustering after the Sybil attack	46

LIST OF GRAPHS

S. No.	Title	Page No.
1.	Cluster head Selection vs Node ID at high transmission range	49
2.	Cluster head Selection vs Node ID at moderate transmission range	49
3	Cluster head Selection vs Node ID at low transmission range	50

List Of Tables

S. No.	Title	Page No.
1.	MANET Characteristics	1
2.	Dimensions of Sybil attack	26
3.	Simulation parameters	36
4.	Observation Table	47
5.	Inference Table	50

Abstract

Mobile ad hoc networks (MANET) is a self-configuring and infrastructure-less network of mobile devices. Since the topology of this network is dynamically changing and there is no central management so it becomes difficult to implement security and this gives an opportunity to intrude into the network and exploit the weaknesses of the same.

Sybil attack is one such attack which takes the advantage of characteristics of MANET and disrupts the network. In a Sybil attack malicious node illegitimately claims multiple identities. This attack has the ability to seriously disrupt various operations like data aggregation, voting, fair resource allocation scheme, misbehavior detection and routing mechanisms etc. In addition to launching a Sybil attack in this project we also aim to disrupt the head selection algorithm of highest degree clustering protocol. We aim to introduce a malicious node into the network known as Sybil node which would slowly take over as the head of the cluster by impersonating itself. Once the Sybil node becomes the head of the cluster, it can disrupt various operations and gain unfair amount of resources which would render the network ineffectual. In this project we visualize how a MANET works and the way clusters are formed. In addition we see how the communication between the mobile devices occur and how the head of a cluster is selected. Once a MANET is established we introduce a Sybil node which would impersonate itself and unsettle the system.

CHAPTER 1

INTRODUCTION

1.1 Introduction

1.1.1 Mobile Ad hoc Network (MANET)

A MANET is a type of ad hoc network in which nodes are in motion and they configure the network time to time. Wireless connections are put up to connect the network [1]. The medium may be Wi-Fi connection, or another medium, such as a cellular or satellite transmission. MANET may operate in limited area say a local area of wireless devices (such as a group of PC), while others may be connected to the Internet. Like in a VANET (Vehicular Ad Hoc Network) vehicles send and transmit messages with the equipments on the roadside .

1.1.1.1 General Characteristics of a Mobile Ad hoc Network (MANET)[1]

Characteristic	Description
Dynamic Topology	The motion of the nodes provide a dynamic topology.
Distributed Operation	.Each node controls the network operation in spite of the central network.
Multi hop routing	The nodes out of reach of each other can communicate via neighboring nodes.
Limited weight terminals	Each node is provided CPU capability along with battery and memory.
Shared physical medium	The medium is accessible to all the nodes that have appropriate equipment.

Table 1: MANET Characteristics

1.1.2 Sybil Attack

Sybil attack was first introduced by J. R. Douceur. According to him, the Sybil attack is an attack in which a single node can rule in the system by presenting multiple fake identities and behaving like multiple legitimate nodes[8].

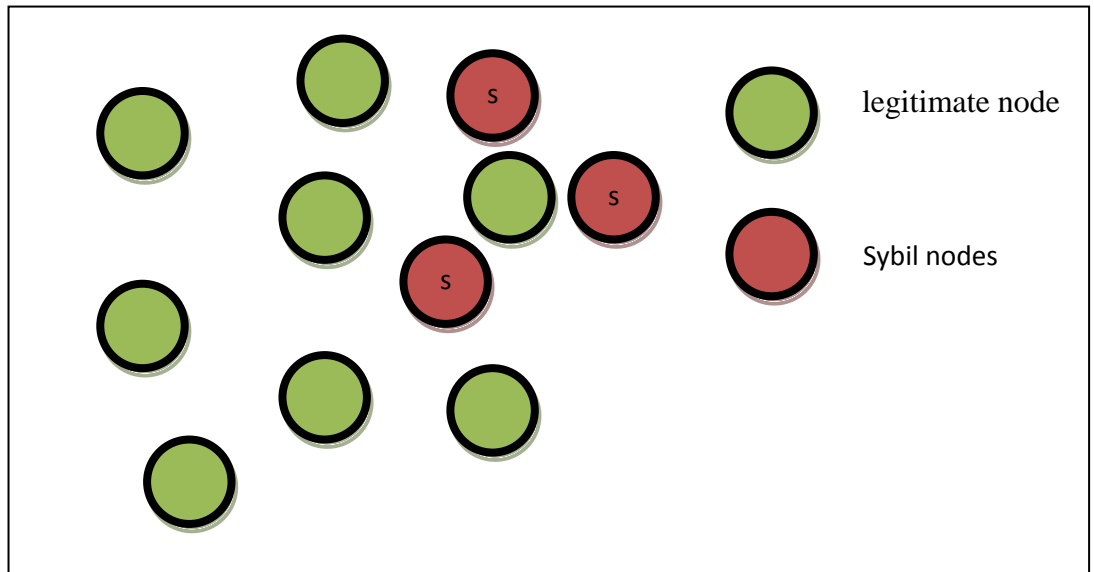


Figure 1: Sybil Nodes [4]

In a MANET, the only way for an entity to know its neighbors is by transmitting hello messages over a shared broadcast communication channel. By taking the advantage of this feature, a malicious node can send messages with multiple fake identities. The node spoofing the identities of the nodes is called malicious node, and the nodes whose identities are spoofed are called Sybil nodes. Figure 1 represents a malicious node along with its three Sybil nodes. If this malicious node communicates with any legitimate node by presenting all its identities, the legitimate node will have illusion that it has communicated with three different nodes. But in actual, there exists only one physical node with multiple different IDs.

Initially malicious node study the behavior of the network and selects the target node among the legitimate nodes. The selection of the target node is an important task in attack because the malicious node will impersonate the ID of the target node. Basically there are 3 types on nodes that are Cluster Head(CH), Ordinary node(ON) and Gateway nodes(GN). All the nodes are initially ordinary nodes which become gateway nodes if they become part of multiple clusters. Gateway nodes are used for inter cluster communication. Each cluster has its leader

Cluster Head(CH) which can be chosen on various parameters.

1.1.2.1 Dimensions of Sybil Attack

Sybil attack goes broadly into 3 dimensions communication, participation and identity[9].

I. Communication

-Direct Communication

- In direct communication legitimate nodes communicate with Sybil nodes directly.

- Indirect Communication

- One or more of the malicious nodes claims to be able in reach with the Sybil nodes.
- Messages sent to a Sybil node are routed via one of these malicious nodes in its way.
- Malicious nodes pretends to forward the message to a Sybil node.

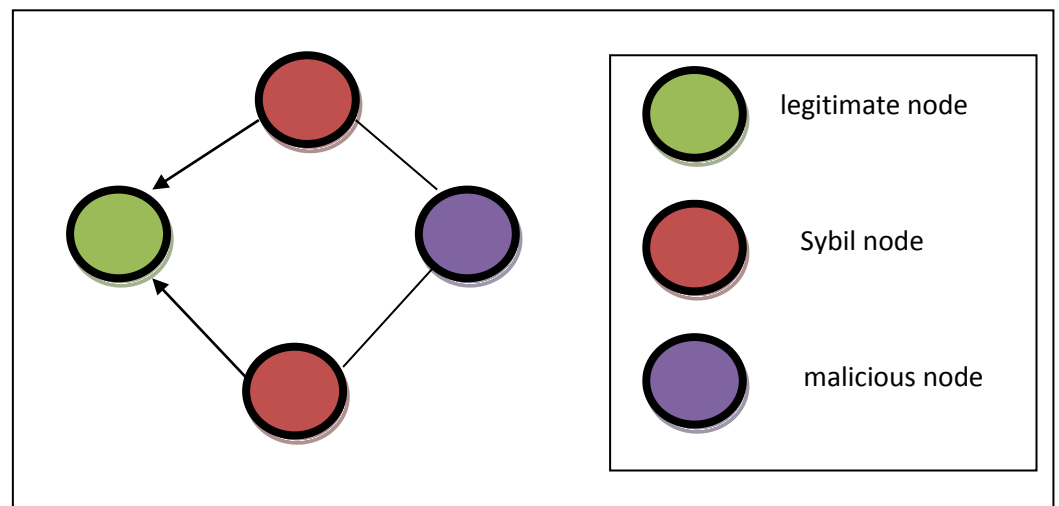


Figure 2: Direct Communication

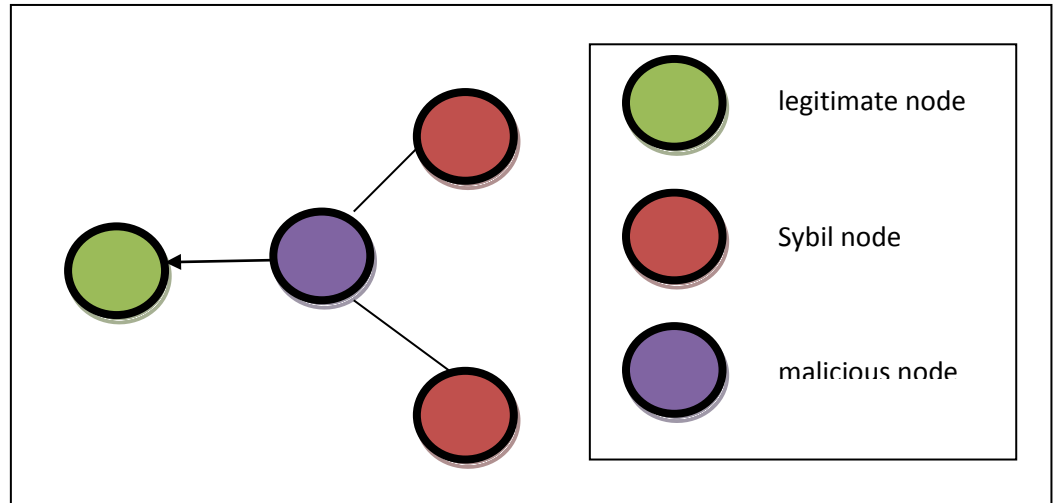


Figure 3: Indirect Communication

II. Participation

- Simultaneous

- The attacker tries to have his Sybil identities all participate in the network at once.
- Hardware entity cycles through identities to make it appear that they are all present simultaneously.

- Non Simultaneous

- Attacker creates a large number of identities from time to time, while only acting as a smaller number of identities at any given time.
- The attacker does this by leaving and joining the network with different IDs.

III. Identity

- Fabricated Identity

- The attacker creates arbitrary new identities.

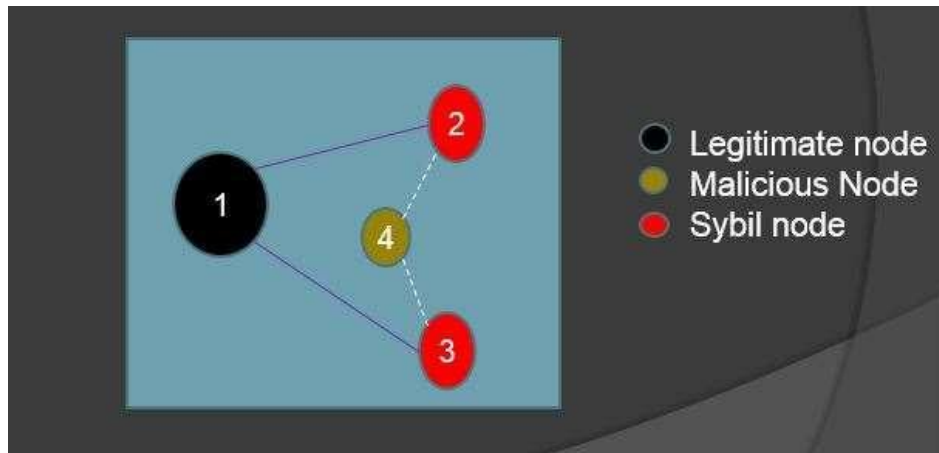


Figure 4: Fabricated Identity

- Stolen Identity

- Sybil nodes are assigned legitimate identities.
- The attack may go undetected if the attacker destroys or disable the original identity.

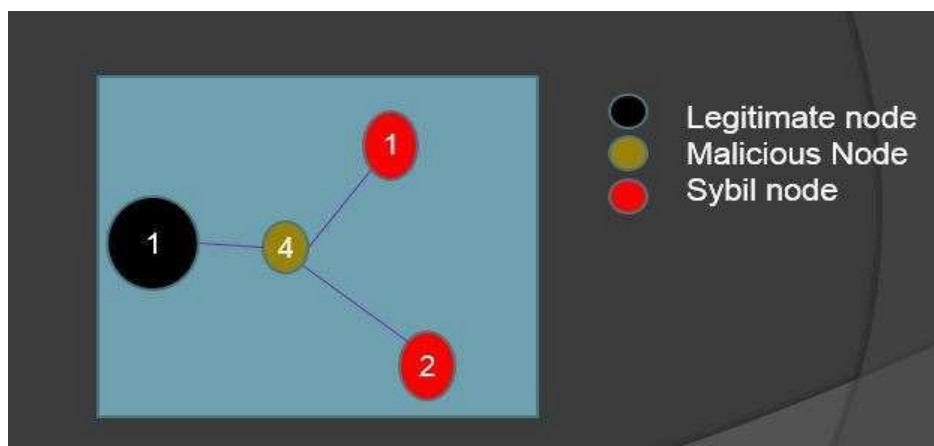


Figure 5: Stolen Identity

1.1.2.2 Sybil attack effect

When a single malicious node is able to convince its neighbor nodes of it as multiple identities, then the system is affected severely. By becoming the cluster head nodes can take over the control of much of the network. Once the Sybil attack succeeds, it has the ability to open the doors for many other attacks also making the system insecure[9].

I. Resource allocation

Sybil attack creates influence in resource distribution in the network. On network there is limited shared resources such as bandwidth here, Sybil node acts as a greedy node, use identity of legitimate user and use network resources on behalf of them. For example channel bandwidth is assigned to each node per time slot basis. Sybil node gains a disproportional amount of resources and create resource availability issue for other nodes in the network.

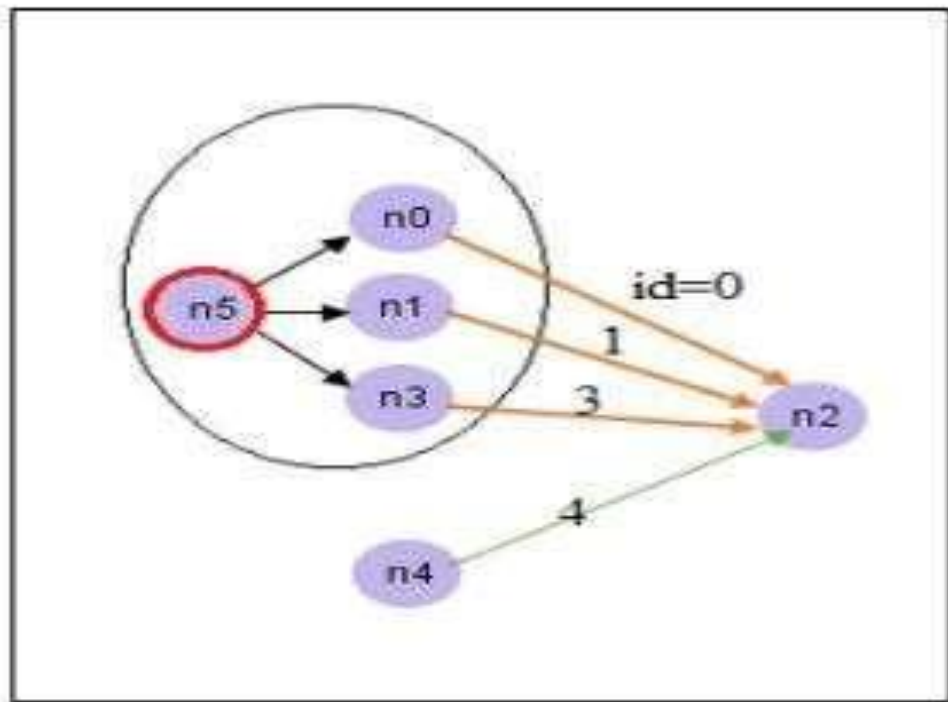


Figure 6: Sybil attack in resource allocation [1]

II. Voting

Sybil node has multiple identities, so in voting Sybil node behave as a selfish node and actively participate in voting such as the cluster head selection, reporting malicious node (IDS). Sybil node sends vote with spoofed identity and impact on

the voting process. It not only impacts voting, but also give explore to the malicious node to behave and control sink hole /cluster head. It leads to poor performance of complete segment and wrong selection of the head.

III. Distributed Storage

Distributed system required redundancy of data. Network node needs to write process data to multiple location for redundancy and distribution of data. Sybil node claims multiple identity in distributed system and other node writes data to Sybil node, assuming writing it at different location.

In figure 6 process id p0, p1 of node n4 and n5 respectively need to write data at n1, n2, and n3 storage location. Sybil node n0 claim fake identity and data stored only at n0. So the data are located at false location and no redundancy is implemented

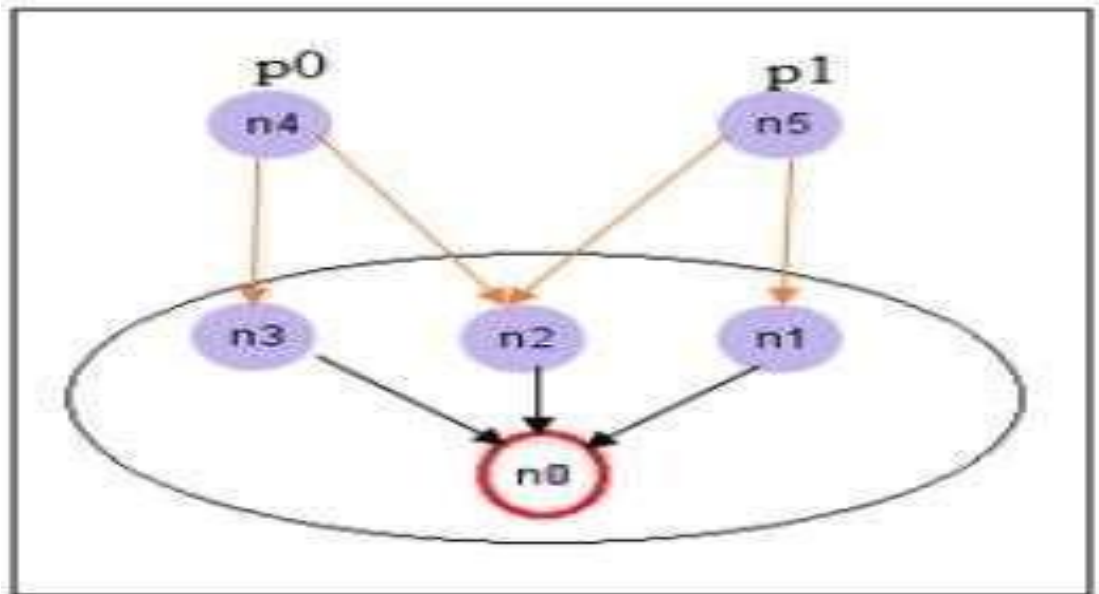


Figure 7: Sybil attack on distributed storage [2]

IV. Data Aggregation

In a special purpose application collection of data is required from different sources (node) in the network. The application requires periodic update data from sensors and monitoring system. Sybil node can send data with clammng identities and create false data sets. This also can lead to a series of chain failures as this data set may be used by further more applications or even the same application to work upon other things.

V. Routing

A routing protocol is used to discover the route from source to destination, Sybil attack may give a big impact on the functionality of the network. Here a malicious node can present multiple fake identities which can involve into a multiple path to disrupt the routing procedure. The complete phenomena will lead to wrong route selection and poor network performance. In fig 6 node Sybil node n5 show identity of n1 and n6 and create an ambiguous routing path in the network.

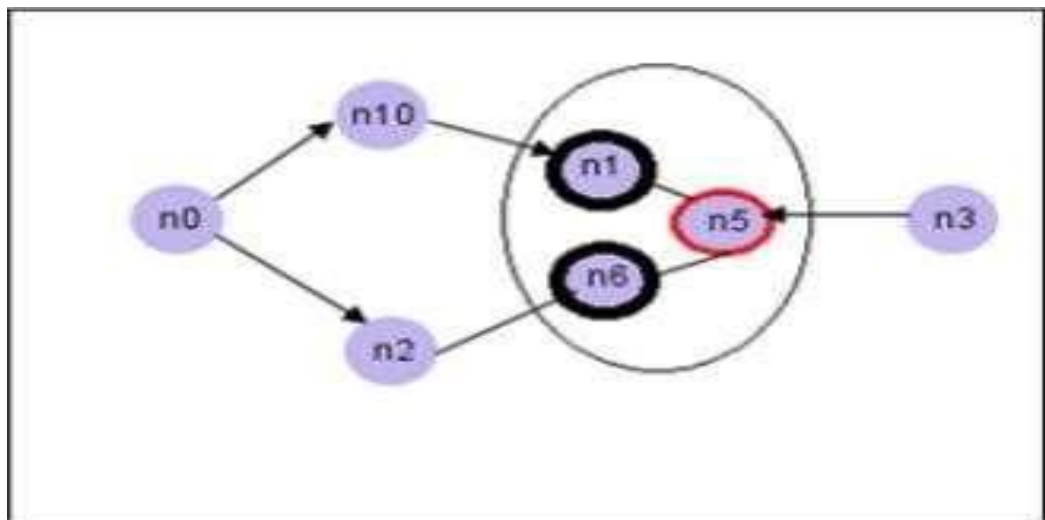


Figure 8: Sybil attack effect on Routing [3]

VI. Hidden Node

Sybil node increases its reputation, trust and credibility, so accuracy for detecting a malicious node is reduced. It creates false alarm generating in (IDS/IPS) system . Attack such as DOS attacker, worm, gray hole are done by other malicious node in the network. The attacker is hidden and its identity remains undetected.

1.1.3 Highest Degree clustering Algorithm

In the highest degree, clustering algorithm, a node with the highest degree is chosen as a cluster head[4]. Cluster is a collection of nodes in a particular area so that the nodes are within the transmission range are placed together. The head of a particular cluster is the node which has the highest degree i.e. Number of nodes connected to other nodes should be maximized. Each node broadcasts its id to the nodes that are within its transmission range. The node with maximum number of neighbor's (i.e., Maximum degree) is chosen as a cluster head.

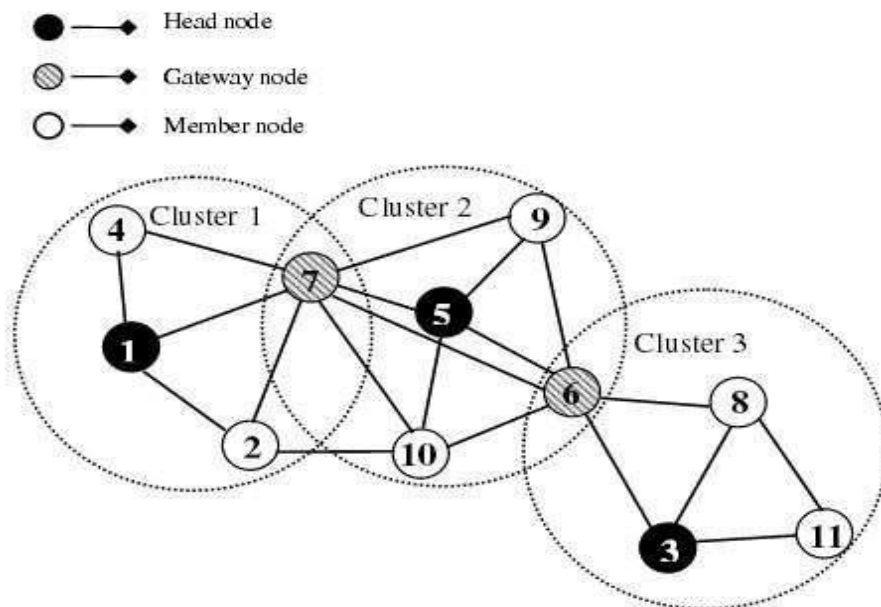


Figure 9: Highest degree clustering algorithm [4]

The neighbors of a cluster head become members of that cluster and can no longer participate in the election process. Since no cluster heads are directly linked, only one Cluster head is allowed per cluster. Any two nodes in a cluster are at most two-hops away from the cluster head is directly linked to each of its neighbors in the cluster. Basically, each node either becomes a cluster head or remains an ordinary node.

1.1.4 Sybil Node Attacks

I. Smurf attack

Sybil node sends ICMP echo message to broadcast address of the network with spoofed IP addresses of the legitimate node, then ICMP echo reply message from all host in network is designated to legitimate node which create flooding traffic for legitimate node

II. Fraggle Attack

Sybil node broadcast UDP echo packet to network with spoofed identity of legitimate node and the legitimate node flooded with reply traffic from the network.

III. ARP-poisoning attack

Sybil node sends a reply message of up request with spoofed IP addresses of gateway so it becomes a gateway for node and perform active or a passive attack on the packet which are send by nodes.

IV. Routing Loop

Sybil node sends a RREQ packet in the network, then nodes send RREP to Sybil node, but the RREQ source not listed in network so RREP packet propagate in whole network until the ttl value of the IP packet is expired.

1.2 PROBLEM STATEMENT

Mobile ad Hoc NETWORK (MANET) has certain characteristics which makes it vulnerable to external attacks which can disrupt the network. Sybil attack is one such attack which utilizes the benefit of MANET characteristics and disrupts the network.

In this project one needs to implement impersonation based Sybil attack on a mobile Ad Hoc network to disrupt the highest degree clustering algorithm. This requires to build a system where a network would be virtually established and the nodes will interact with each other as in the case of the Mobile ad Hoc network. Once the network is established and the heads are being chosen as per the highest degree, clustering algorithm a Sybil attack needs to be launched. This would require to place a malicious node which would further create the Sybil nodes by impersonating its id. This would lead to disruption of highest degree, clustering algorithm effecting the system capabilities.

1.3 OBJECTIVES

- To analyze Mobile ad Hoc Network attacks and their impact on the quality of service.
- To investigate the Sybil attack and its effect on MANET.
- To propose an algorithm for highest degree, clustering algorithm in MANET.
- To perform a Sybil attack in the MANET and disrupt the highest degree clustering algorithm.
- To study the effects and consequences of a Sybil attack.
- To perform the implementation in Advanced Java Based Environment using swings, servlets etc.

1.4 METHODOLOGY

This will cover the detailed explanation of methodology that is being used to make this project complete and working well. The method is used to achieve the objective of the project that will accomplish a perfect result. In order to evaluate this project, the methodology based on System Development Life Cycle (SDLC), generally three major steps, which is planning, implementing and analysis.

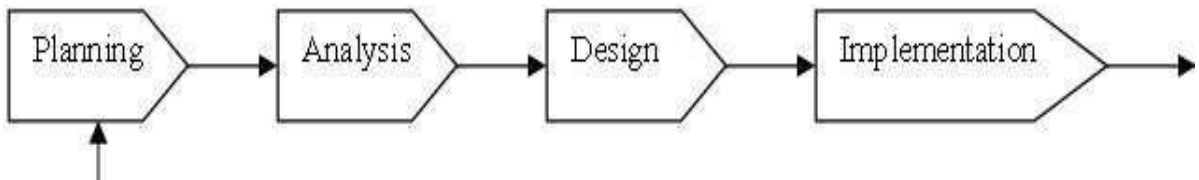


Figure 10: SDLC Life Cycle

This final year project used three major steps to implement project starting from planning, implementing and testing. All the methods used for finding and analyzing data regarding the project related.

I. Planning

- Data collection
- Software and Hardware requirement

II. Implementing

- Testing point
- Implement the project

III. Analysis

- Analyze the performance
- Identify the conclusion

1.5 ORGANIZATION

Chapter 1 highlights and underlines the MANET characteristics. In this chapter various security issues with MANET are discussed and then the various attacks which utilize them are discussed. The key focus, however remains on how to perform an impersonation based Sybil attack so that the network could be disrupted.

The detailed literature review from the research paper, books, journals and conferences are done in **Chapter 2**. In this chapter, the extracts from assorted research papers on MANET and Sybil attacks are taken and depicted.

Chapter 3 covers the system development which is the key aspect of this work. In this chapter, the proposed model, algorithm and related parameters are emphasized.

The simulation of implementation results with the relative performance analysis is shown in **Chapter 4**. In this chapter, the simulation results and screenshots are revealed to depict and defend the proposed work.

Chapter 5 ends with the detailed conclusion and scope of the future work which guides the upcoming students and research scholars to enhance the current work with higher efficiency and effectiveness.

CHAPTER 2

LITERATURE SURVEY

For complete, justification and solving the problem definition, a number of research papers, magazines, journals and online links were investigated in details. In this chapter, the details of research papers and journals are specified from where we have analyzed the content and formulated the problem.

A number of research scholars and scientists have written a number of research papers and found excellent results. This section underlines all those research papers and their extracts.

2.1 Study of MANET: Characteristics, Challenges, Application and Security Attacks by Aarti and Dr S S Tyagi [1]

In this research paper, we studied about the characteristics of the mobile ad hoc Network that are listed below. We also studied about the advantages and application of MANET and also get to know its vulnerabilities..

Characteristics of MANET

- 1. Distributed operation:** There is no background network for the central control of the network operations, the control of the network is distributed among the nodes. The nodes involved in a MANET should cooperate with each other and communicate among themselves and each node acts as a relay as needed, to implement specific functions such as routing and security.
- 2. Multi hop routing:** When a node tries to send information to other nodes, which is out of its communication range, the packet should be forwarded via one or more intermediate nodes.

3. **Autonomous terminal:** In a MANET, each mobile node is an independent node, which could function as both a host and a router.
4. **Dynamic topology:** Nodes are free to move arbitrarily with different speeds; thus, the network topology may change randomly and at unpredictable times. The nodes in the MANET dynamically establish routing among themselves as they travel around, establishing their own network.
5. **Light-weight terminals:** In maximum cases, the nodes at MANET are mobile With less CPU capability, low power storage and small memory size.
6. **Shared Physical Medium:** The wireless communication medium is accessible to any entity with the appropriate equipment and adequate resources. Accordingly, access to the channel cannot be restricted.

Advantages of MANET

1. The advantages of an Ad-Hoc network include the following:
2. They provide access to information and services regardless of geographic position.
3. Independence from central network administration. Self-configuring network, nodes are also act as routers. Less expensive as compared to wired networks
4. Scalable-accommodates the addition of more nodes
5. Improved flexibility
6. Robust due to decentralize administration.
7. The network can be set up at any place and time.

MANETs Applications

Some of the typical applications include:

1. **Military battlefield:** Ad-Hoc networking would allow the military to take advantage of commonplace network technology to maintain an information network between the soldiers, vehicles, and military information head quarter.
2. **Collaborative work:** In some business environments, the need for collaborative computing might be more important outside office environments than inside and where people do need to have outside meetings to cooperate and exchange information on a given project.
3. **Local level:** Ad-Hoc networks can autonomously link an instant and temporary multimedia network using notebook computers to spread and share information among participants at a e.g. Conference or classroom. Another appropriate local level application might be in home networks where devices can communicate directly to exchange information.
4. **Personal area network and Bluetooth:** A personal area network is a short range, localized network where nodes are usually associated with a given person. Short-range MANET such as Bluetooth can simplify the intercommunication between various mobile devices such as a laptop, and a mobile phone.
5. **Commercial Sector:** Ad hose can be used in emergency/rescue operations for disaster relief efforts, e.g. Fire, flood, or earthquake. Emergency rescue operations must take place where non-existing or damaged communications infrastructure and rapid deployment of a communication network is needed.

MANETs Vulnerabilities

Vulnerability is a weakness in security system. A particular system may be vulnerable to unauthorized data manipulation because the system does not verify a user's identity before allowing data access. MANET is more vulnerable than wired networks. Some of the vulnerabilities are as follows:

1. **Lack of centralized management:** MANET doesn't have a centralized monitor server. The absence of management makes the detection of attacks difficult because it is not easy to monitor the traffic in a highly dynamic and large scale ad-hoc network.
2. **No predefined Boundary:** In mobile ad-hoc networks, we cannot precisely define a physical boundary of the network. The nodes work in a nomadic environment where they are allowed to join and leave the wireless network. As soon as an adversary comes in the radio range of a node, it will be able to communicate with that node.
3. **Cooperativeness:** Routing algorithm for MANETs usually assumes that nodes are cooperative and non-malicious. As a result a malicious attacker can easily become an important routing agent and disrupt network operation.
4. **Limited power supply:** The nodes in mobile ad-hoc network need to consider restricting power supply, which will cause several problems. A node in mobile ad-hoc network may behave in a selfish manner when it is finding that there is only limited power supply

2.2 Literature Survey on latest research issues in MANET by Mrs Padma and Mr R. Suresh [2]

MANET ARCHITECTURE

The nodes in a MANET can be classified by their capabilities. A Client or Small Mobile Host (SMH) is a node with reduced processing, storage, communication, and power resources. A Server or Large Mobile Host (LMH) is a node having a larger share of resources. Servers, due to their larger capacity to contain the complete DBMS and bear primary responsibility for data broadcast and satisfying client queries. Clients typically have sufficient resources to cache portions of the database as well as storing some DBMS query and processing modules. As both clients and servers are mobile, the speed at which the network topology changes can be rapid. A variety of techniques have been proposed to assist in the routine tasks of the MANET. New protocols were necessary as the protocols for fixed infrastructures and static networks do not perform well when node mobility is included. A global routing structure is also not useful in MANET due to its dynamic topology and need for distributed control. Work on routing is ongoing and is coordinated through the Internet Engineering Task Force (IETF).

In Figure 1, a few nodes of a MANET are shown graphically. It is important to note that each node has an area of influence. This is the area over which its transmissions can be heard. At LMH will initially have a larger area of influence as it generally has a more powerful battery. As the power level decreases, the area of influence of any node will shrink. This is due to the fact that the power available to broadcast is reduced.

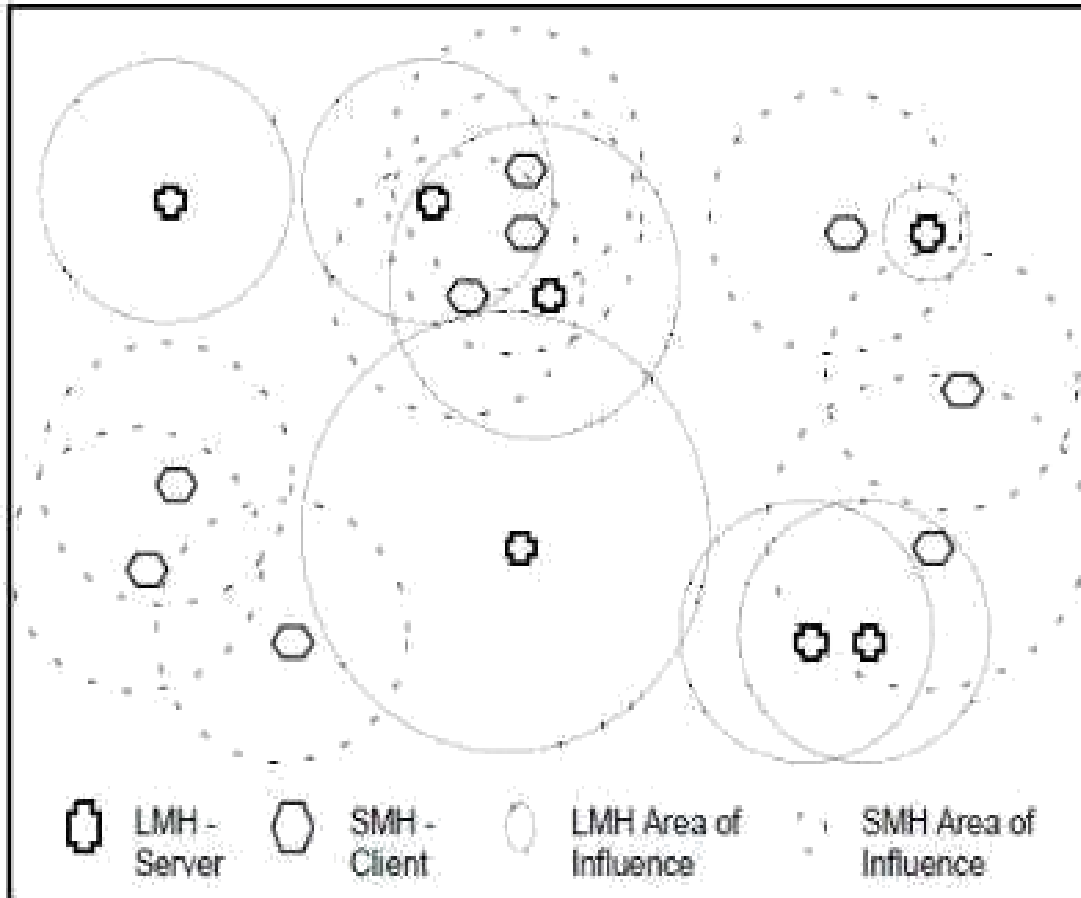


Figure 11: Nodes of a MANET [2]

Network nodes may operate in any of three modes that are designed to facilitate the reduction in power used

1. **Transmit Mode:** this is the mode using the most power. It allows both the transmission and reception of messages and consumes 3000 to 3400 mW .
2. **Receive Mode:** the CPU is capable of processing information and is also capable of receiving notification of messages from other nodes and listening to broadcasts. 1500 to 1700 mW are consumed in this mode.

3. **Standby Mode:** the CPU does no processing and the node has no ability to send/receive messages. The node is inactive and consumes only 150 to 170 mW. This mode allows a node to turn itself off for short periods of time without requiring power-up or initialization. A node with no remaining power, or one that is off, is not currently a part of the network.

2.3 A Review of Routing Protocols for Mobile Ad-Hoc Networks (MANET) by Alex Hinds, Michael Ngulube, Shaoying Zhu, and Hussain Al-Aqrabi [3]

MANET ROUTING PRINCIPLES

1. **Proactive Routing:** Proactive protocols rely upon maintaining routing tables of known destinations, this reduces the amount of control traffic overhead that proactive routing generates because packets are forwarded immediately using known routes, however routing tables must be kept up to-date; this uses memory and nodes to periodically send update messages to neighbors, even when no traffic is present, wasting bandwidth. Proactive routing is unsuitable for highly dynamic networks because the routing tables must be updated with each topology change, this leads to increased control message overheads which can degrade network performance at high loads.
2. **Reactive Routing:** Reactive Protocols use a route discovery process to flood the network with route query requests when a packet needs to be routed using source routing or distance vector routing. Source routing uses data packet headers containing routing information meaning nodes don't need routing tables. However, this has high network overhead. Distance vector routing uses next hop and destination addresses to route packets, this requires nodes to store active route information until no longer required or an active route timeout occurs, this prevents stale routes.

Types of MANET Routing

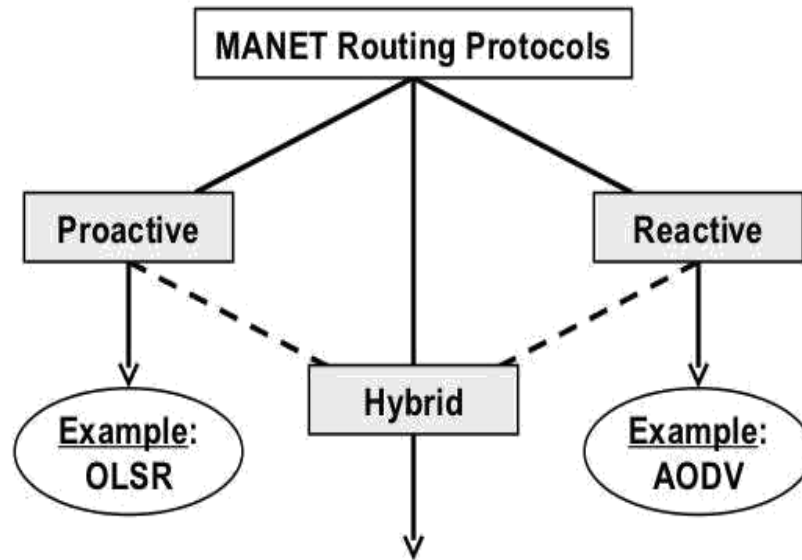


Figure 12: Types of routing protocols [3]

3. **Hybrid Routing:** Hybrid protocols combine features from both reactive and proactive routing protocols, typically attempting to exploit the reduced control traffic overhead from proactive systems whilst reducing the route discovery delays of reactive systems by maintaining some form of the routing table.

Performance information about the different protocols is very limited and no details of any testing methodologies is provided, because of this the validity of some claims made cannot be verified.

2.4 Clustering & Cluster Head Selection Techniques in Mobile Adhoc Networks by V. Preetha and Dr K. Chitra [4]

In this paper, we studied the role of a cluster Head. We saw the responsibilities of the cluster head. We also studied the different ways to choose a cluster head, which can be classified in the following.

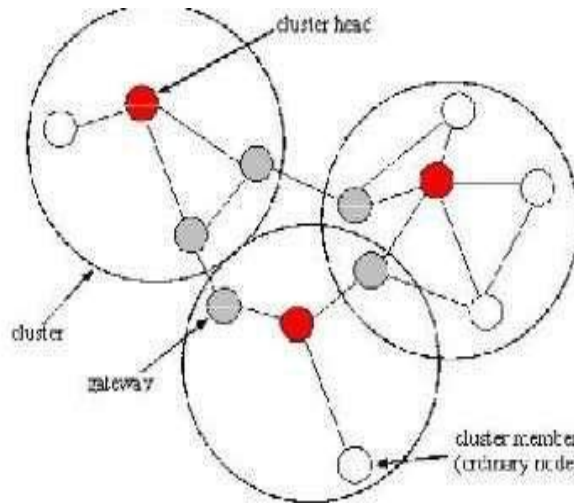


Figure 13: Cluster structure in MANET [4]

We studied multiple ways in which a cluster head can be chosen. They can be classified as Identifier-based, Connectivity based, Mobility based, Cost based, Power based algorithms as shown below:

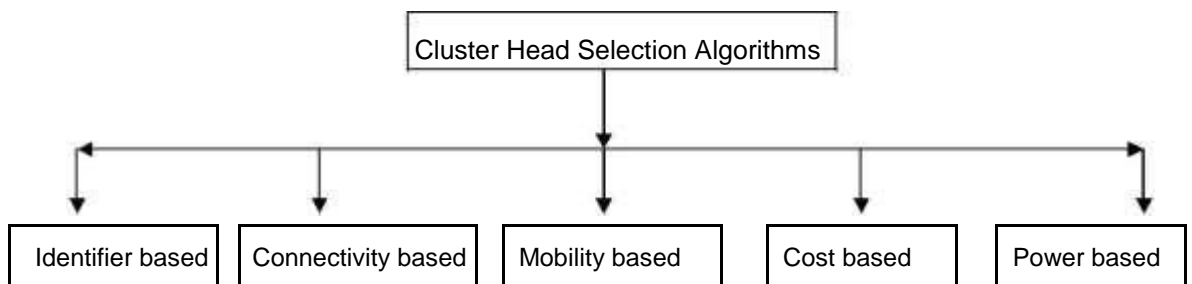


Figure 14: Cluster Head Selection Algorithm[4]

1.) Identifier based

In this we studied about the Lowest ID cluster algorithm (LIC). Every node is given a unique ID which they broadcast to all their neighbors. In LIC the clusters are formed beginning from the lowest id node. In the cluster the node

having the least id is the cluster head. The selection of the cluster head is done periodically.

2.) Connectivity-based clustering

In this we studied about the Highest Degree clustering algorithm (HDC).

The main objective of the highest node degree, clustering is to minimize the no of clusters formed. Clusters are formed and the node having the highest degree is chosen as cluster head in the respective cluster. If the multiple nodes are having the same degree then the node with lowest id is selected as cluster head.

3.) Mobility-aware clustering

Classification: Mobility-based d-hop clustering algorithm.

Procedure: Local stability is computed in order to select some nodes as cluster heads. A node may become a cluster head if it is found to be the most stable node among its neighbors. Thus, the cluster head will be the node with the lowest value of local stability among its neighbors.

4.) Low cost of maintenance clustering

Classification: Least cluster change algorithm (LCC)

Procedure: The cluster formation simply follows LIC, i.e. Initially mobile nodes with the lowest ID in their neighborhoods are chosen as Cluster heads.

In this adaptive clustering scheme, every mobile node i keep its own ID and the ID of its direct neighbors in a set G_i . Each mobile node with the lowest ID in their local area declares to be a cluster head and set its own ID as its cluster ID (CID). The CID information includes a mobile node's ID and CID. When a mobile node i receives CID information from a neighbour j , it deletes j from its set G_i .

5.) Power-aware clustering

Classification: Power-aware connected dominating set

Procedure: In this scheme Energy level (el) instead of ID or node degree is used to determine whether a node should serve as a cluster head.

2.5 SYBIL ATTACK ON LOWEST ID CLUSTERING ALGORITHM IN THE MOBILE ADHOC NETWORK by Amol Vasudeva and Manu Sood [5]

A MANET is a collection of mobile nodes interconnected via wireless links. Any infrastructure or centralized administration is not needed. The motion of the nodes provides a dynamic topology. Nodes within transmission range can do direct communication. In the MANET reconfiguration of nodes is done to form small clusters. Every node can communicate within the cluster. Every cluster chooses a CH on the basis of node characteristics. Gateway nodes are used for inter cluster communication and ordinary nodes are used to communicate within the cluster. Due to nodes mobility, limited bandwidth, high error rates, limited battery power and continuously changing topology, its routing protocols are intricately designed. Many researchers have categorized the routing protocols under proactive, reactive and hybrid. In Proactive protocols, regular exchange of network topology packets is done by the nodes to keep the record of routing of other nodes in the network. In the case of routing protocols, discovery of the routes is done on demand by flooding the packets into the network. Hybrid protocols are the mixture of both proactive and reactive protocols.

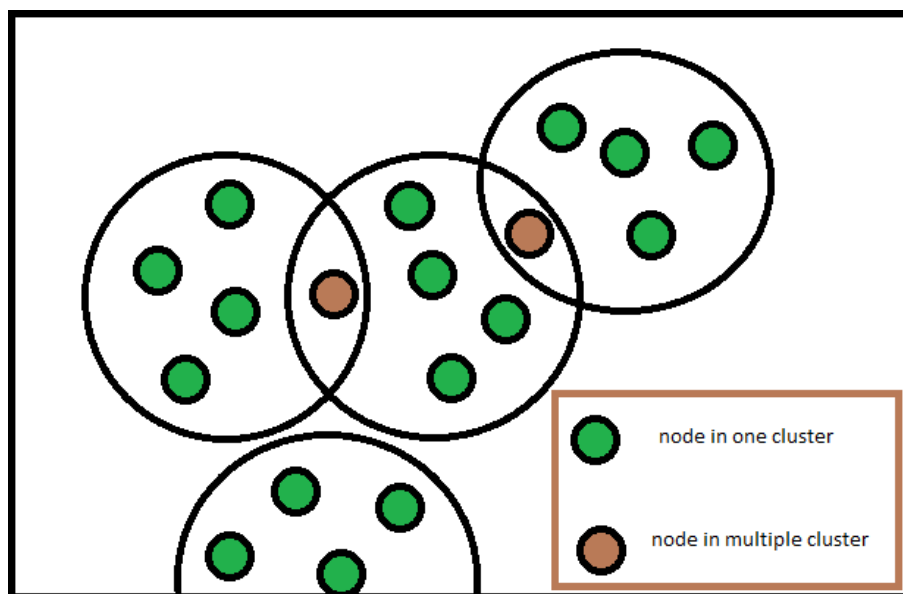


Figure 15: Nodes type in MANET

MANET is more vulnerable to various attacks. The vulnerabilities are the 4 L's of MANET. The lack of centralized management makes tough detection of various attacks tough as no monitoring is done on node movements. Cooperative routing protocol assumes that all nodes are legitimate. So malicious node can easily become the part of the network. No predefined boundary in MANET provides a nomadic environment in which nodes can enter and exit the network anytime. The selfishness of the nodes due to limited power supply undone the job of finding the malicious nodes. In multi-path routing the message the Sybil nodes may create

multiple faked identities in intermediate of the communication path. In the second case, malicious nodes may create faked nodes in the cluster of the legitimate nodes. The malicious nodes will never know that they are communicating to the malicious node in real.

A Sybil attack is an attack in which a malicious node illegally claims multiple identities by impersonating other nodes or by claiming fictitious identities. Sybil attacks are also capable of disrupting the routing mechanisms in mobile ad hoc networks. In this paper we have shown how the cluster head selection based on the highest degree, clustering algorithm is being disrupted by this attack. It is the first attempt to show the vampire act of Sybil attack on the highest node degree clustering algorithm. To do impersonation, we have introduced a Sybil attack in which the malicious node varies its transmit power to create a number of illegitimate node i.e. Sybil nodes. These nodes will then communicate with the legitimate node. When it drains the battery of the legitimate node, it steal node id and attack destructively.

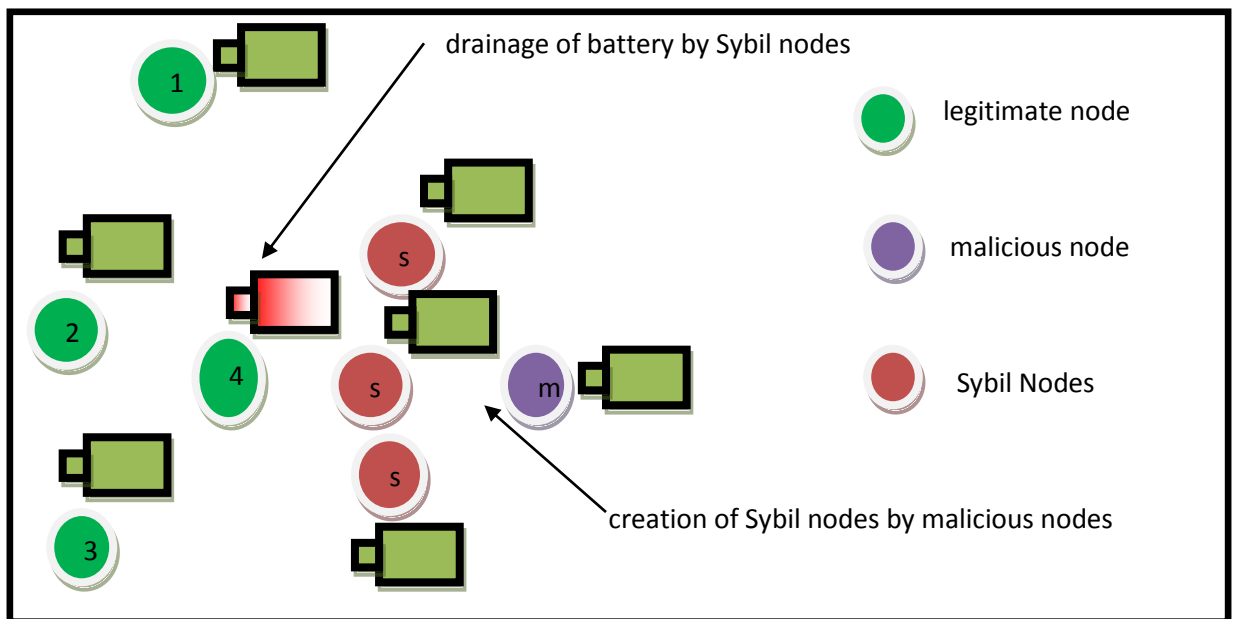


Figure 16. Nodes in MANET

Sybil attack

Sybil attack was first introduced by J. R. Douceur [8]. According to Douceur, a single entity rule the most in the system by presenting multiple faked identities in Sybil attack[8].

Dimensions of Sybil Attack

A Sybil attack can be represented using the three dimensions that are Communication, participation and identity.

Dimension	Types	
Communication	Direct Communication	Indirect Communication
	Sybil nodes communicate with the legitimate nodes directly.	Sybil nodes communicate with the legitimate Node via malicious node.
Participation	Simultaneous Participation	Non Simultaneous
	Attacker participates with all his identities at once.	Attacker participates with a large number of identities over a Period of time.
Identity	Fabricated Identity	Stolen Identity
	The attacker creates arbitrary new identities for Sybil nodes.	Attacker assigns legitimate identities to Sybil nodes.

Table 2. Dimensions of Sybil attack

Effects of Sybil Attack--

As per Newsome et al in [7], the Sybil attack can disrupt the following mechanism:

- **Resource Allocation--** Sybil nodes acts as a greedy node in the resource allocation and get more share of resources. For instance, suppose resource distribution is on the basis of number of IDs. So Sybil node will get disproportional resources by creating multiple fake identities.
- **Voting--** Sybil node acts as a selfish node and create multiple fake voters to increase its votes in the election of cluster head. When it becomes the cluster head, then all messages pass through it and gets what's happening in the simulation environment.

- **Distributed Storage**-- Sybil nodes with its fake multiple ids gets maximum of data which is assumed to be with multiple legitimate nodes but is with the single Sybil nodes.
- **Data Aggregation**--Lots of false data sets are created with the multiple ids of the Sybil nodes, which is assumed to be from many legitimate nodes.
- **Routing**--it disrupts multiple routing paths by creating multiple fake identities. As it behaves to be in many parts so it gets many of the transmitted messages.

CHAPTER 3

SYSTEM DEVELOPMENT

3.1 Algorithm for Sybil attack in the highest degree clustering

- All the nodes identify their respective one-hop neighbors by broadcasting Hello messages.
- A node with maximum node degree among its neighboring nodes is elected as a cluster head.
- In the case of a tie between two or more nodes in terms of node degree, the node with minimum identity is elected as a cluster head.
- The nodes within the range of a cluster head are called the member nodes.
- The remaining nodes that are not in the cluster are once again scanned and the process repeats until all nodes are assigned to a cluster.
- Nodes within the range of two or more clusters are called the gateway nodes.

3.1.1 Algorithm for Impersonation Based Sybil Attack Disrupting the Highest Degree Clustering

Assumptions

1. Let N be the number of nodes in a Mobile Ad Hoc Network, at any instant of time, with their IDs as

$$x_i^j : i = 1, 2, \dots, N.$$

2. One of the nodes in the network, say x_m , is a malicious node.

3. The malicious node x_m is capable of introducing its Sybil nodes by varying the transmission power.

Steps

1. After behaving normally for some time, this malicious node will gain the access to the necessary information about the network including its neighbors and their IDs.
2. In the next step the malicious node x_m generates an S number of Sybil nodes that can communicate with the legitimate neighboring nodes. Let the degrees of Sybil nodes are

$$n_j : j = 1, 2, \dots, S \text{ where } S < N.$$

The degrees are chosen such that $n_j < n_i$.

3. Now, in addition to itself, the malicious node will also include its Sybil nodes in the selection of cluster head. Since the degrees of all the Sybil nodes are less than the degrees of all other legitimate nodes in the networks, the legitimate node with the highest degree will become the cluster head, repeatedly. In addition, the Sybil attacker node x_m will also use its Sybil nodes to communicate again and again using its different IDs so as to keep the head node busy all the time, until its battery is drained, completely.
4. After the battery of this cluster head node is drained completely, the malicious node can impersonate its ID and assign it to one of its Sybil nodes to make it a cluster head.

3.2 Problems with the existing System

The problem with this system is that target node may move out of the range of the malicious node. Thus, it is necessary for the malicious node to move in the direction of the target node. For this the malicious node can be equipped with the localization scheme to track the position of the target node.

The other problem is that the battery of the malicious node will also get drained after a particular period of time, due to regular communication with the target node. Thus, a malicious node is required to be equipped with the more resources in terms of memory, battery power and processing power.

As an alternative, once the battery of the malicious node has been drained to a certain threshold level, this malicious node can be substituted by another fresh malicious node. In doing so, the previous malicious node will have to transfer its complete information to the new node before being getting disabled.

3.3 Software Development model

Prototype model is used for software development as our application is interactive and prototype model is best suited for such applications.

Advantages of Prototype model:

- Users are actively involved in the development
- Since in this methodology a working model of the system is provided, the users get a better understanding of the system being developed.
- Errors can be detected much earlier.
- Quicker user feedback is available leading to better solutions.
- Missing functionality can be identified easily

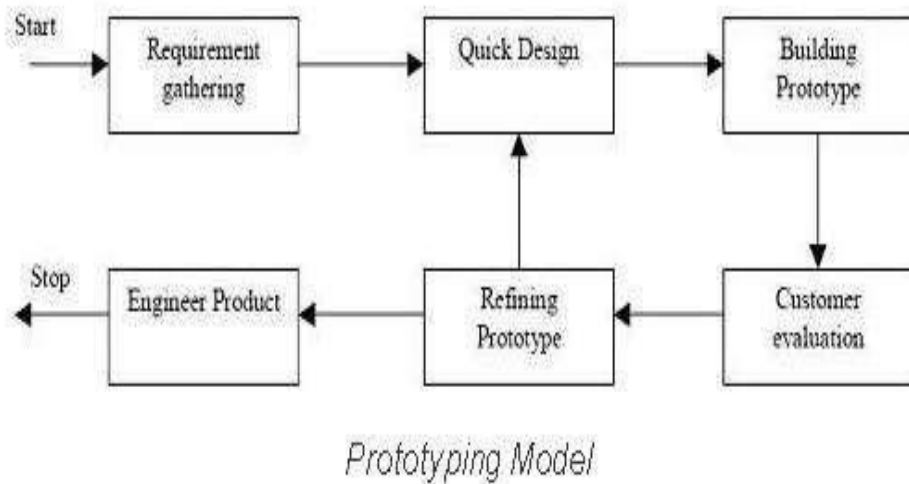


Figure 17: Prototype Model

3.4 Diagrams

3.4.1 Data Flow Diagram

Zero level DFD

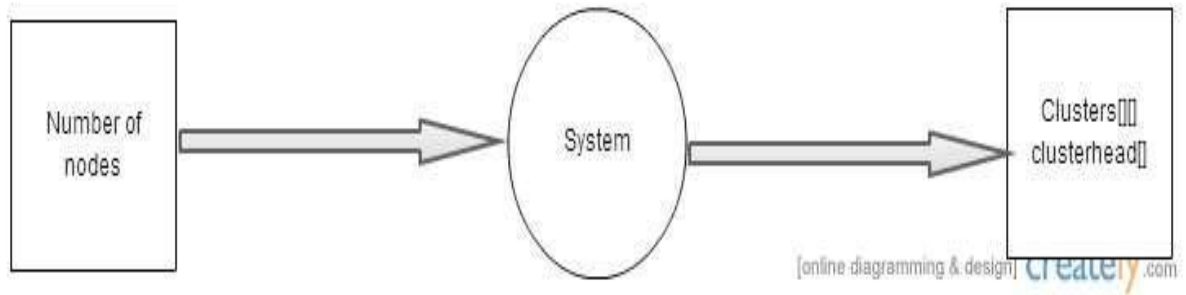


Figure 18: Zero level DFD

First level DFD

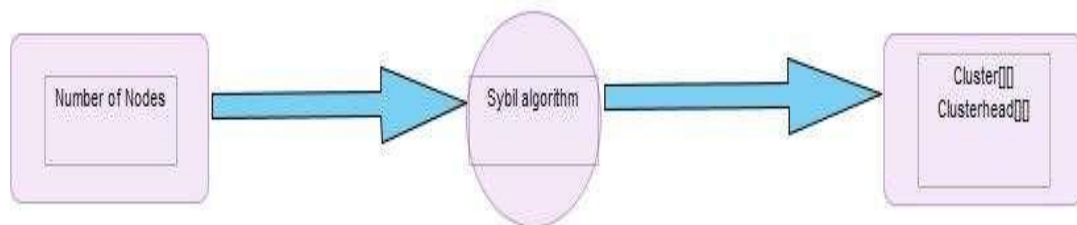


Figure 19: First level DFD

Second level DFD

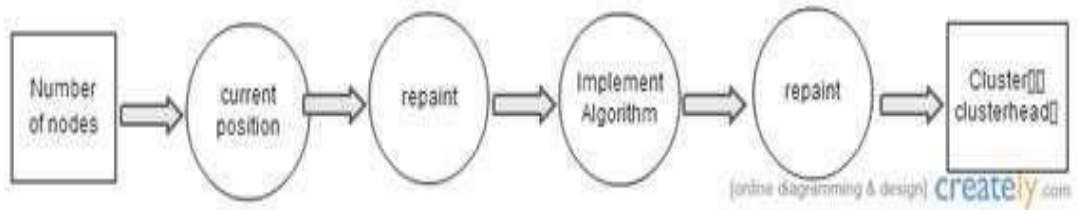


Figure 20: Second level DFD

3.4.2 Use Case Diagram

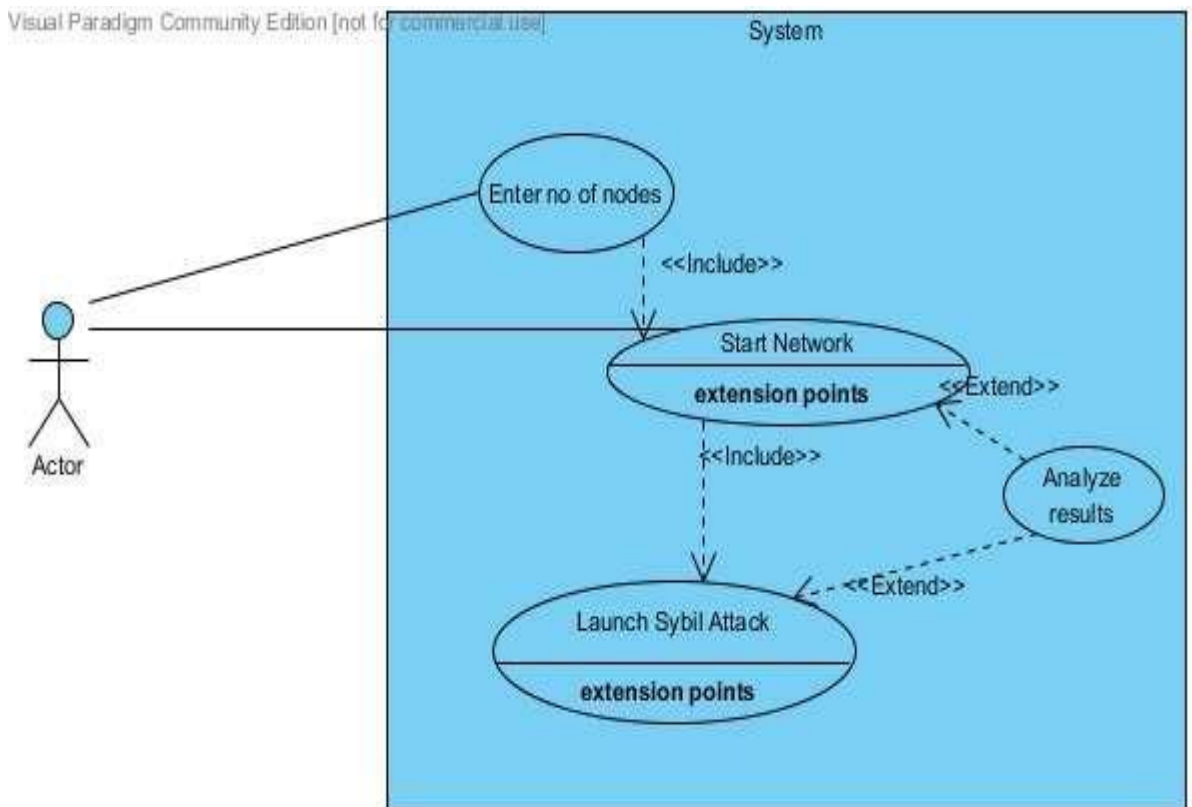


Figure 21: Use Case Diagram

3.5 HARDWARE REQUIREMENTS

The minimum requirements needed to perform operations are

- Intel Pentium Processor at 2 GHz or Higher
- RAM 256MB or more
- Hard disk capacity 10GB or more

3.6 SOFTWARE REQUIREMENTS

The software required to perform the implementation are

- Windows or Linux Operating System (Ubuntu, Fedora)
- JDK 8 and above
- Eclipse / NetBeans IDE
- Dia - The Diagram Editor

CHAPTER 4

PERFORMANCE ANALYSIS

4.1 Complexity of Algorithm To make adjacency matrix.

- Time Complexity: $O(n^2)$
- As for every n nodes we will check $(n-1)$ nodes, whether they can communicate or not, so it will be $O(n^2)$.

To find the node with the maximum degree

Time Complexity: $O(n)$

To form clusters

I. Best Case

When all the nodes are in a single cluster.

- Time Complexity: $O(n)$

II. Worst Case

- When all the nodes have degree 0 and each form its own cluster.
- $n+(n-1)+(n-2)+(n-3)+\dots+1 = (n(n+1))/2 = O(n^2)$
- First node checks for n nodes to make its cluster, 2^{nd} for $n-1$ and so on.
- Time Complexity: $O(n^2)$

4.2 Outputs at various stages

4.2.1 Simulation Environment Parameters

Summary of the Simulation Environment	
Parameter	Value
Simulator	Java
Simulation area	300 pixel x 300 pixel
Transmission range of legitimate node	120 pixels
Number of legitimate nodes	20
Total number of observations	25
An observation period	Variable, by selecting the pause button
Mobility model	Random Waypoint Model
Speed of nodes	0-10 pixels/second
Movement direction	$0-2\pi$
A number of malicious nodes	1
Number of Sybil nodes	5
Presentation of Sybil nodes	Simultaneous
Transmission range of Sybil nodes	30,25,20,15,10
	50,40,30,20,10
	100,80,60,40,20

Table 3:Simulation Parameters

4.2.2 Home Page

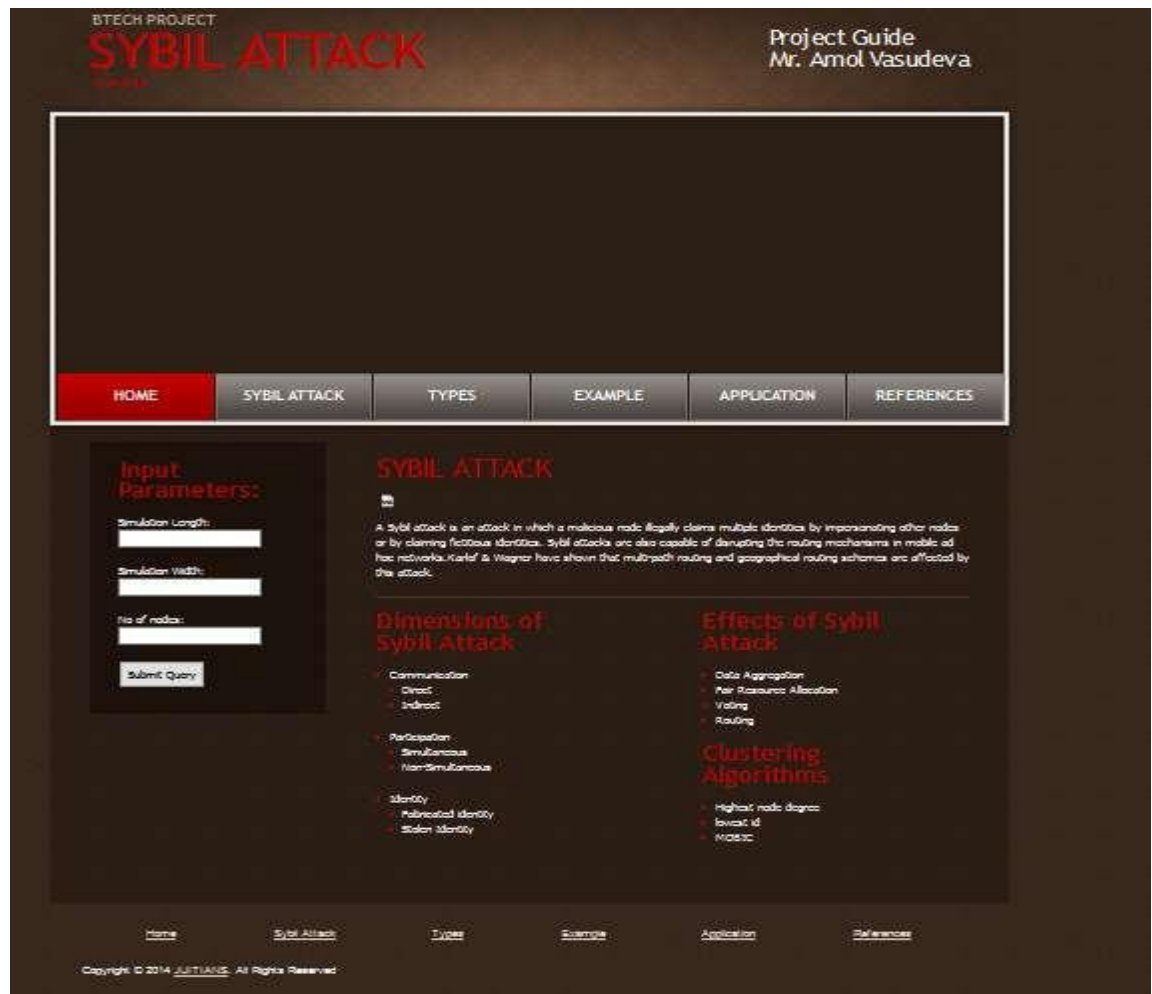
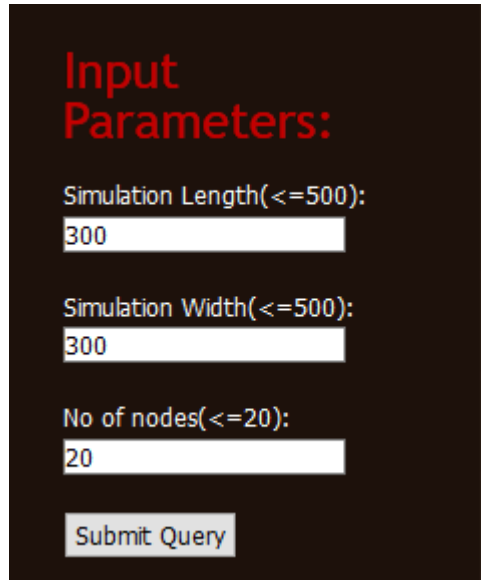


Figure 22: Home page of application

We have created an application in which the homepage will be shown to the user whenever the user will run the Java code. On this homepage the user will have information about the whole app and can understand the Sybil Attack. The Home page has an input form which the user has to fill to set the parameters for the simulation area. The input parameters will include the length, width of the simulation area and the number of nodes.

4.2.3 Input page



Input Parameters:

Simulation Length(<=500):
300

Simulation Width(<=500):
300

No of nodes(<=20):
20

Submit Query

Figure 23: Input page

When the user will enter the parameters and click on the submit button. Then the Simulation

The area with the movements of all the nodes will be shown to the user. Initially, all the nodes will be legitimate nodes. In this case the simulation area will be about 300 x 300 pixels and number of nodes will be 20 initially. The transmission range is taken 120 for the legitimate node when the area is 300 x 300 pixels. The Simulation environment has a constraint that it can be exceeded by 500 x 500 and nodes cannot exceed 20. It is done to show all the parameters of the node in the single JFrame.

4.2.4 Output with all legitimate nodes

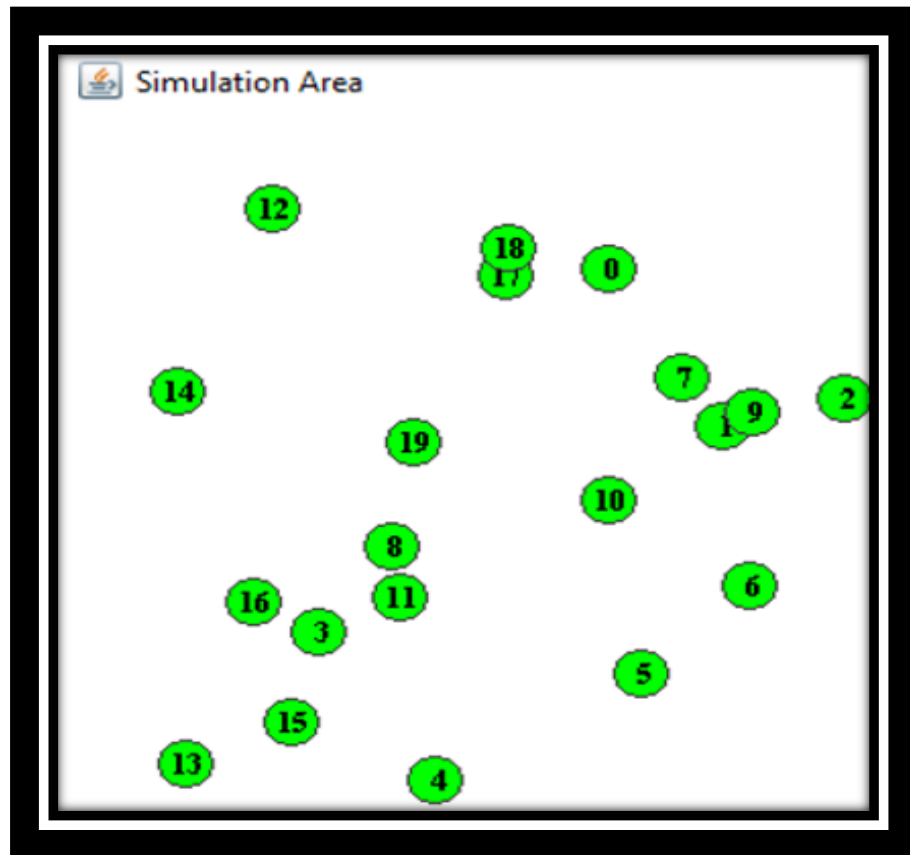


Figure 24: Output with all legitimate nodes

All the nodes shown in the green color represent legitimate nodes.

For now we are taking the starting positions randomly. All the nodes move within the rectangular area with varying speeds and randomly changing directions.

4.2.5 Output (Node Positions)

Users will be shown the current location of all the nodes, that is x and y coordinates of all the nodes which will be selected randomly.

The position will be changing randomly and the changes will be reflected accordingly.

NODE PARAMETERS		
Node id	Node x	Node y
0	409	308
1	99	282
2	387	199
3	214	166
4	238	151
5	316	262
6	385	231
7	50	129
8	36	376
9	110	471
10	148	341
11	387	342
12	303	418
13	207	250
14	53	226
15	372	63
16	346	185
17	137	392
18	265	130
19	456	167

Figure 25: current position of all nodes

4.2.6 Output (Adjacency matrix)

The attacker will see the Adjacency matrix and will attack the node having the highest degree.

In the Adjacency matrix we will show 1 for the nodes which are adjacent and 0 if they are not adjacent.

Adjacency Matrix

	N0	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	N11	N12	N13	N14	N15	N16	N17	N18	N19
N0	0	0	0	1	1	0	0	1	0	0	1	0	0	0	1	0	0	1	0	1
N1	0	0	0	0	1	1	1	0	0	1	0	0	1	1	1	1	1	0	1	1
N2	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	1	0	0	1	0
N3	1	0	0	0	1	0	0	1	0	0	1	0	0	0	1	1	0	0	1	1
N4	1	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1
N5	0	1	1	0	0	0	1	1	1	1	0	1	1	1	0	1	0	0	1	0
N6	0	1	1	0	0	1	0	1	1	1	1	1	1	1	0	1	0	0	1	0
N7	1	0	1	1	0	1	1	0	1	1	1	1	0	0	0	1	0	0	1	1
N8	0	0	1	0	0	1	1	1	0	1	1	1	0	0	0	1	0	0	1	0
N9	0	1	1	0	0	1	1	1	1	0	1	1	1	1	0	1	0	0	1	0
N10	1	0	1	1	0	0	1	1	1	1	0	1	0	0	1	1	0	0	1	1
N11	0	0	1	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	1	0
N12	0	1	0	0	0	1	1	0	0	1	0	0	0	1	1	1	1	0	1	1
N13	0	1	0	0	0	1	1	0	0	1	0	0	1	0	0	1	1	0	1	0
N14	1	1	0	1	1	0	0	0	0	0	1	0	1	0	0	1	1	1	0	1
N15	0	1	1	1	0	1	1	1	1	1	1	0	1	1	1	0	1	0	1	1
N16	0	1	0	0	1	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
N17	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
N18	0	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	0	0	0	1
N19	1	1	0	1	1	0	0	1	0	0	1	0	1	0	1	1	0	1	1	0

Figure 26: Adjacency matrix

4.2.7 Highest node degree Clustering

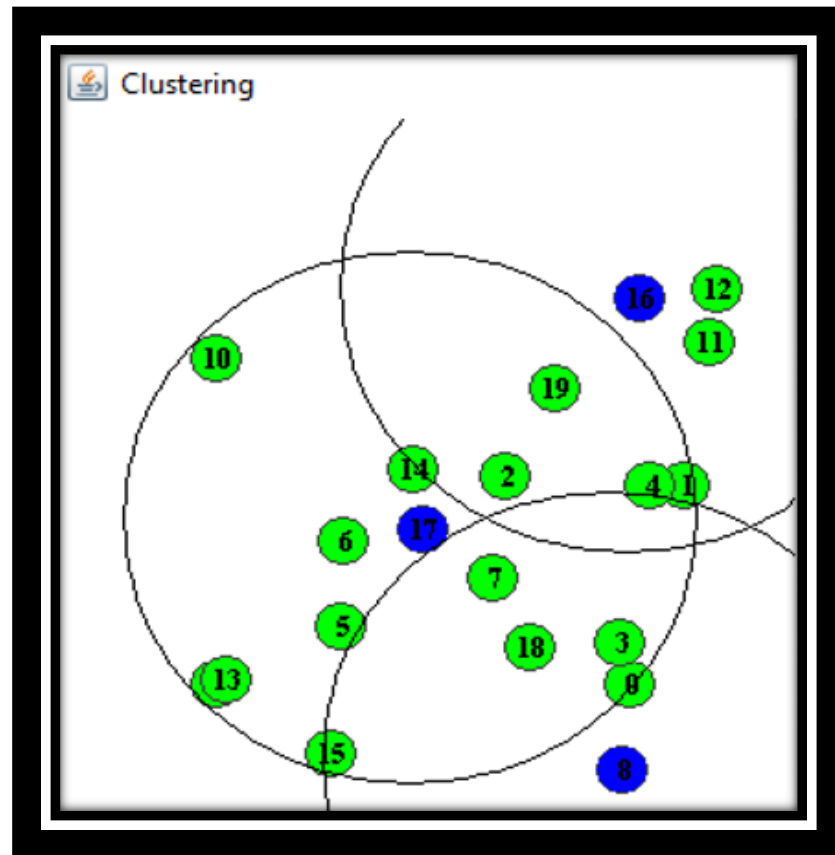


Figure 27: Clustering

Clustering is done on the basis of the highest degree of the nodes. In this case three clusters are shown. Nodes that are in blue color are the cluster head in their clusters.

4.2.8 Output with all communication details

```
Message Details

Node 3 sends Message to Node 7.
Node 5 sends Message to Node 7.
Node 3 sends Message to Node 7.
Node 5 sends Message to Node 7.
Node 2 sends Message to Node 4.
Node11 sends Message to Node 0.
Node 3 sends Message to Node 7.
Node 5 sends Message to Node 7.
Node 13 sends Message to Node 0.
Node 5 sends Message to Node 7.
Node 3 sends Message to Node 7.
Node 5 sends Message to Node 0.
Node 7 sends Message to Node 0.
Node 7 sends Message to Node 0.
Node 12 sends Message to Node 0.
Node 12 sends Message to Node 0.
Node 12 sends Message to Node 0.
Node 12 sends Message to Node 0.
```

Figure 28: Message Details

The message details show that the communication is done with node 0 and node 7 because These are the nodes that are attacked by the Sybil Nodes.

4.2.9 Battery levels of each node

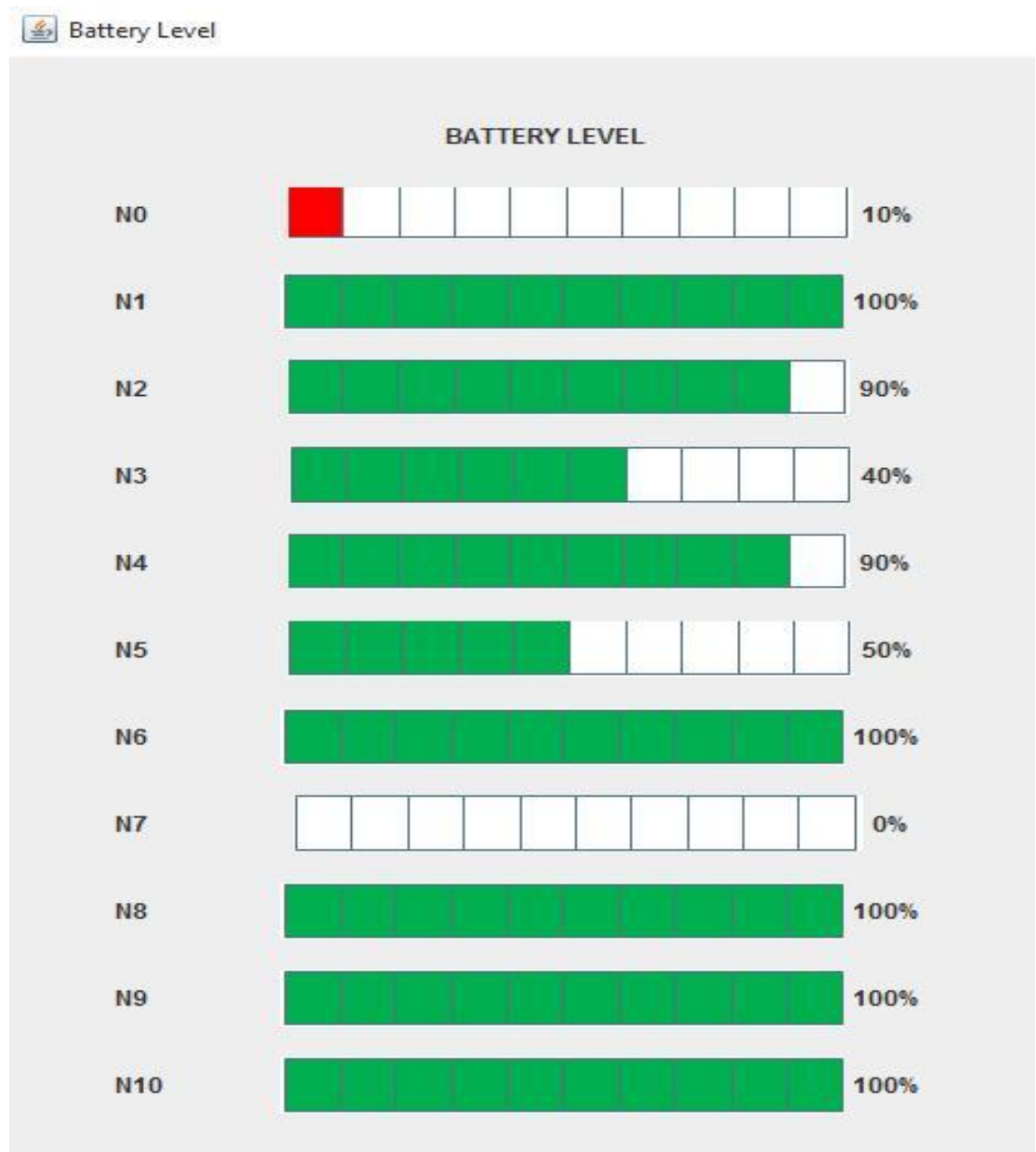


Figure 29: Battery levels

As the communication occurs, battery level decreases, which is depicted in the above shown Format.

4.2.10 Output (Launching Sybil attack)

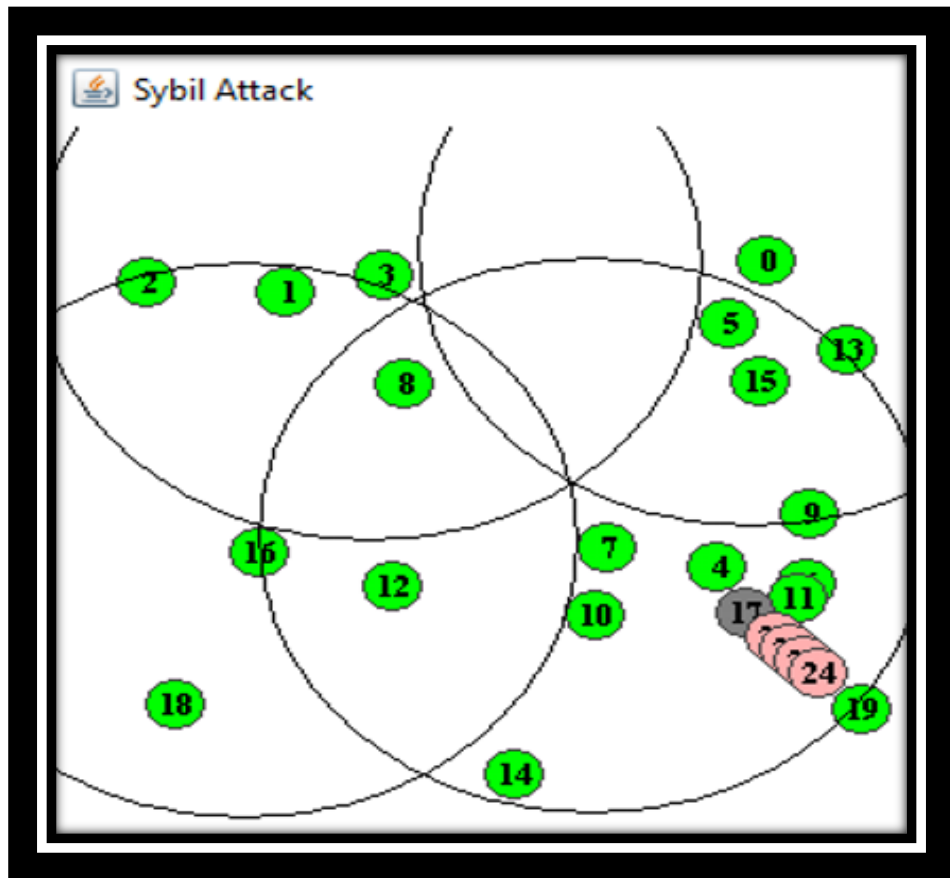


Figure 30: Sybil attack

In this phase malicious nodes select the target node. In this case 17 is the target node which is shown in the gray color. A malicious node creates Sybil nodes, which are shown in pink color.

4.2.11 Clustering after the Sybil attack

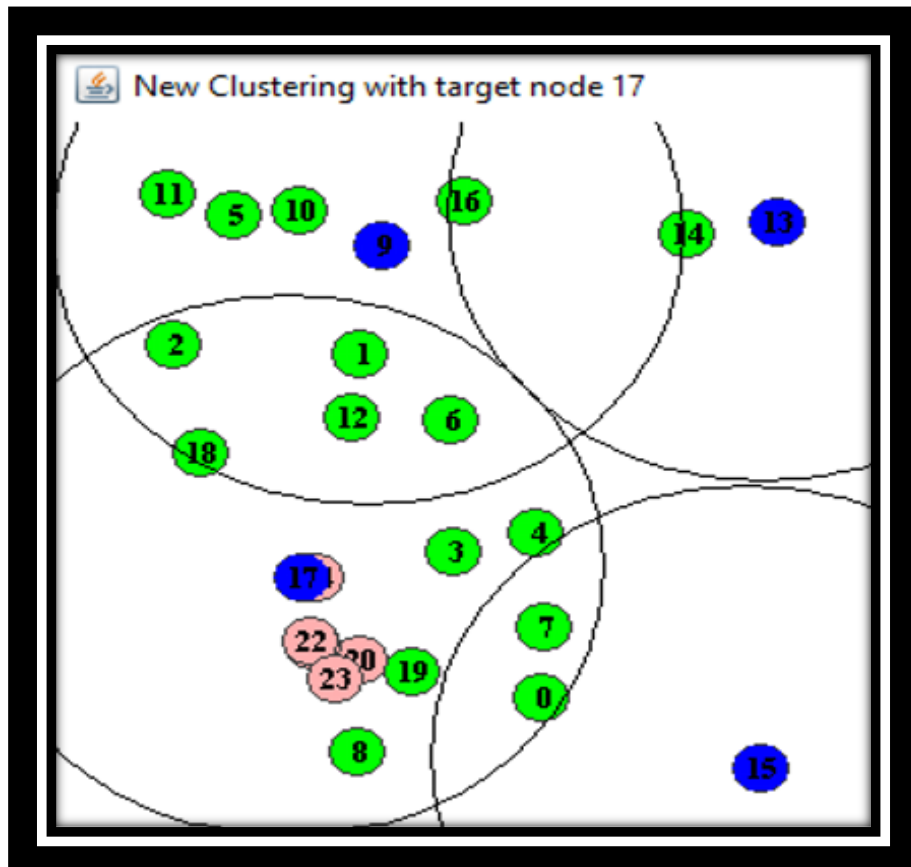


Figure 31: Clustering after the Sybil attack

In this phase malicious nodes make target node cluster head again and again. In this case 17 is the target node. A malicious node creates Sybil nodes, which are shown in pink color which always vote for the target node to make it cluster head again and again.

4.3 Results

4.3.1 Observation Table

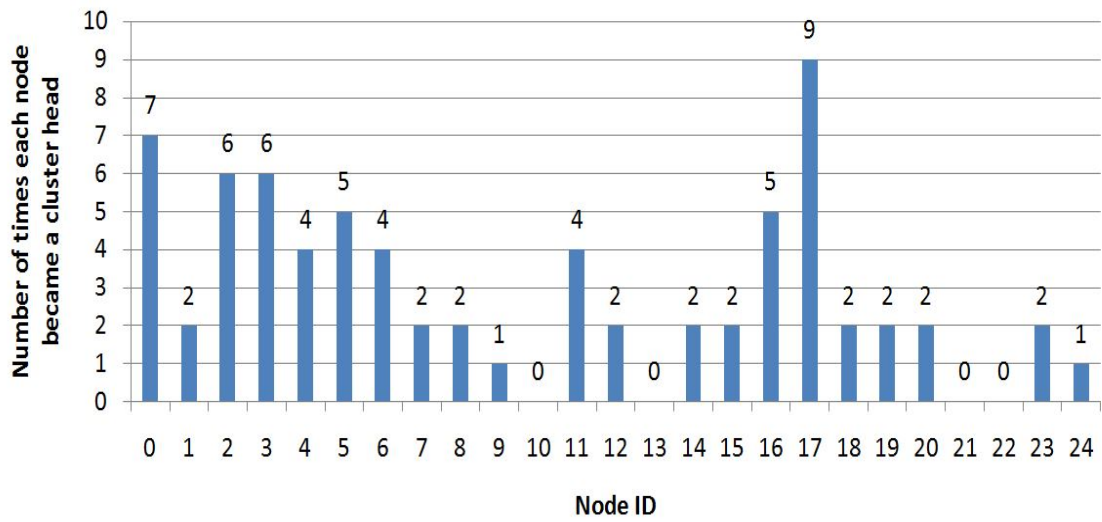
We have done this project to study the effect of the transmission range of the Sybil nodes on the Sybil attack.

Observation No	Number of Cluster formed	Cluster Head {Cluster Members}
1	4	17 {8,10,12,13,16,19,20,21,22,23,24}, 1 {3,5,6,10,11,12,13,14,15,18 }, 4 {0,2,3,5,7,13,16,19}, 9 {6,8,10,12,14,15,18}
2	4	17 {0,1,3,4,6,11,16,19,20,21,22,23,24}, 5 {2,6,7,8,12,13,14,15,18}, 9 {3,11,14,15,18}, 10 {2,7,12}
3	4	17 {1,2,3,5,6,10,11,16,19,20,21,22,23,24}, 0 {2,3,4,6,7,8,14}, 12 {4,5,7,15,16}, 13 {3,6,8,9,18}
4	2	6 {0,1,2,3,5,6,8,10,11,12,14,15,16,17,18,19,20,21,22,23,24}, 5 {1,4,7,10,11,12}
5	2	17 {0,1,2,3,6,7,8,9,10,11,12,13,14,19,20,21,22,23,24}, 4 {12,13,14,15,16,18}
6	3	12 {0,1,2,4,5,7,9,10,11,13,14,15,18,19}, 17 {9,10,13,16,18,19,20,21,22,23,24}, 8 {3,4,9,13,16,18,19}
7	4	17 {9,10,11,13,18,19,20,21,22,23,24}, 2 {0,1,4,5,6,7,12,14}, 16 {4,12,14}
8	3	0 {1,4,7,9,10,11,12,13,14,15,16,18}, 17 {2,3,7,19,20,21,22,23,24}, 5 {4,6,8,11,12}
9	3	8 {0,1,2,4,5,6,7,9,10,11,12,13,15,16,18,19}, 17 {1,2,3,9,16,18,19,20,21,22,23,24}, 14 {0,5,10}
10	4	17 {2,7,9,10,15,18,19,20,21,22,23,24}, 6 {0,1,3,8,9,10,11,13,14,16,18}, 12 {2,4,7,21,22}, 5 {4,14}
11	3	17 {0,6,7,9,14,16,18,19,20,21,22,23,24}, 11 {1,3,7,8,9,10,13,14,15,19,20}, 12 {4,8}

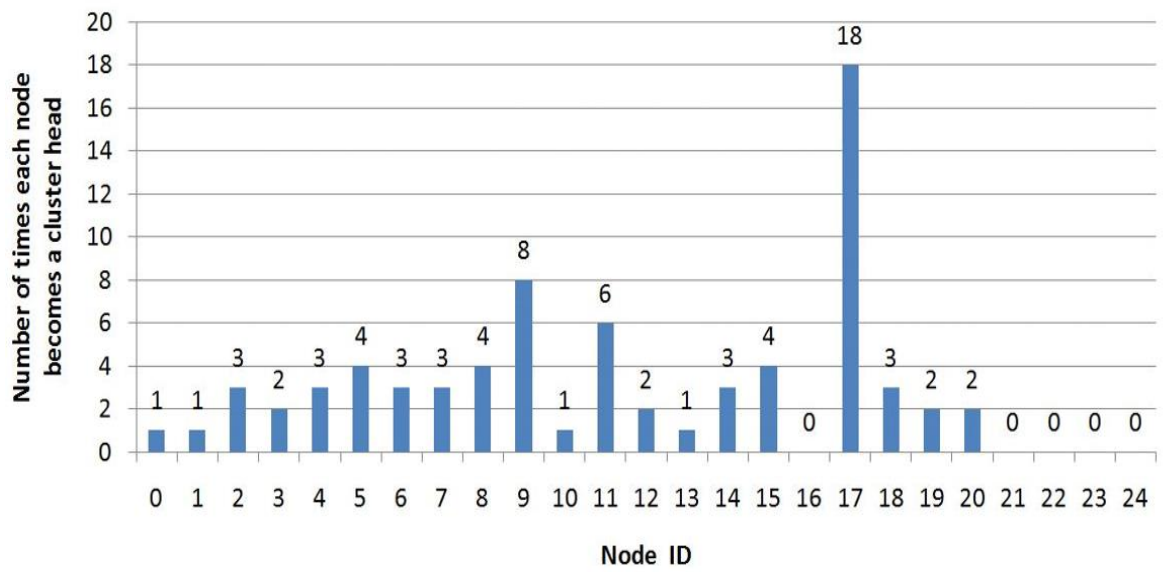
12	3	17 {6,7,8,12,14,15,19,20,21,22,23,24}, 16 {1,3,5,6,8,11,13,15,18,19}, 0 {3,6,8,9,10,15,18}
13	4	17 {1,12,13,14,16,19,20,21,22,23,24}, 9 {1,5,8,10,18,19}, 7 {0,4,12}, 15 {11}
14	3	17 {3,8,9,10,11,16,19,20,21,22,23,24}, 6 {0,1,2,3,11,14,18,24}, 15 {1,3,4,5,7,12,13}
15	3	6 {0,2,3,4,5,7,8,11,13,14,16,18}, 17 {2,4,7,10,12,19,20,21,22,23,24}, 15 {1,7,9,12}
16	4	5 {2,3,4,6,7,9,11,12,13,14,15,17,18,19}, 10 {2,3,7,8,11,15,19,22,23}, 16 {6,9,13,14,18}, 1 {0,2,7}
17	4	17 {1,8,9,15,18,19,20,21,22,23,24}, 3 {0,1,2,5,6,9,10,11,16}, 14 {4,5,6,10,11,13}, 12 {1,4,7,13}
18	3	17 {2,3,5,6,8,11,12,13,15,19,20,21,22,23,24}, 6 {0,1,4,7,9,12,14,16,18}, 10 {0,2,13}
19	3	17 {3,14,15,16,18,19,20,21,22,23,24}, 5 {0,1,2,3,4,6,7,8,12,14,16}, 9 {8,13,15,18,19,20}
20	4	5 {1,3,4,6,7,8,9,11,12,14,15,18}, 3 {0,2,4,7,10,11,13,15}, 16 {6,8,9,12,4,18}, 17 {19,20,21,22,23,24}
21	4	17 {1,7,9,10,13,14,16,19,20,21,22,23,24}, 18 {4,5,6,11,12,14,16}, 3 {0,6,8,9,13,16}, 15 {2,4,5,12,14,21}
22	3	17 {1,2,6,8,9,10,11,18,19,20,21,22,23,24}, 6 {0,3,5,8,11,12,13,14,15,16}, 1 {0,4,7,12,13,14}
23	4	17 {3,14,15,18,19,20,21,22,23,24}, 8 {1,2,3,4,6,7,10,11,12,14,15}, 9 {4,7,11,12,13,15,18}, 5 {13,15,18}
24	4	17 {1,12,14,15,16,19,20,21,22,23,24}, 18 {0,5,8,9,13,16,19,21}, 4 {2, 6, 7, 10, 12}, 1 {3, 15}
25	3	17 {1,3,6,8,9,11,12,13,19,20,21,22,23,24}, 15 {0,2,4,5,6,7,9,10,13,14,16,18}, 3 {9,14,18}

Table 4: Observation Table

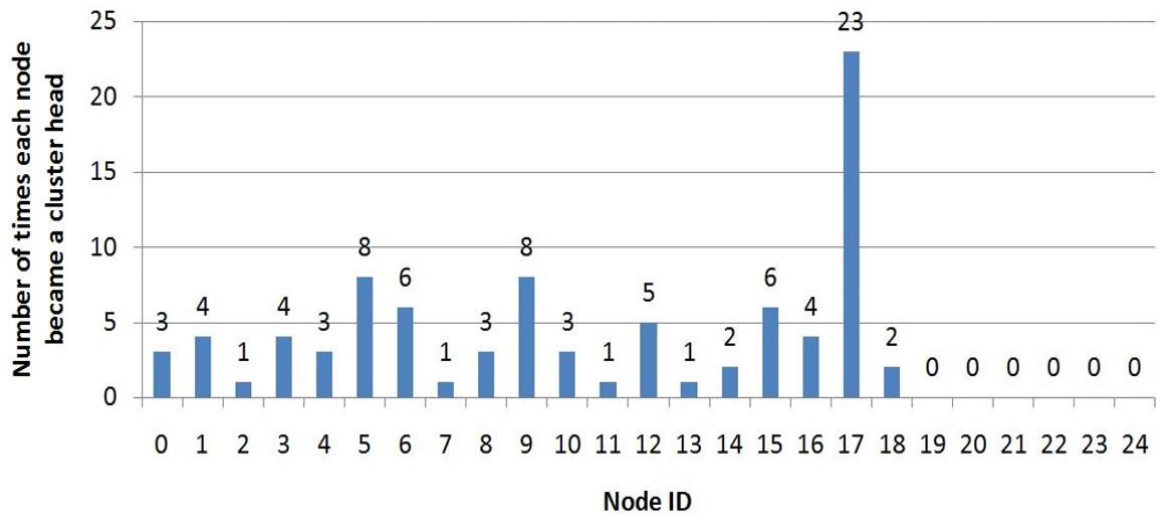
4.3.2 Graph to show the effect of transmission range of Sybil nodes on Sybil attack



Graph 1: Cluster head Selection vs Node ID at High transmission range



Graph 2: Cluster head Selection vs Node ID at moderate transmission range



Graph 3: Cluster head Selection vs Node ID at low transmission range

Transmission Range	No of times target node is chosen as CH(OUT OF 25 obs.)	Percentage of selection of target node to become CH
{10,15,20,25,30}	23	92%
{20,30,40,50,60}	18	72%
{20,40,60,80,100}	9	36%

Table 5: inference Table

It is justified from the observation table that if the transmission range of the target node is the minimum then it is chosen Cluster Head maximum times.

CHAPTER 5

CONCLUSIONS

5.1 Conclusions

MANET is susceptible to many security attack. No centralized identity management in the MANET and the requirement of exclusive and distinctive as well as persistent identity for each node for their security protocol to be viable, Sybil attack propose a dangerous impact to such a network. A Sybil attack is in which a malicious node in the network, illegally claims to have many identities on a single physical device. A Sybil attacker can harm to the ad hoc networks in one or various ways.

For example, a Sybil attacker can interrupt location-based or multipath routing by participating in the routing, giving the fake impression of being legal nodes on different locations or node-disjoint paths.

The highest degree, clustering algorithm makes the node with the highest degree as the head of the cluster and the nodes with it is connected as part of that cluster. Any communication between the two clusters take place through the cluster head and cluster head control the functioning of the entire cluster by acting as the dominant force.

In wireless sensor networks, a Sybil attacker can change the complete aggregated reading outcome by participating many times as a different node. Sybil attack has the ability to disrupt the entire network by taking an inappropriate amount of resources. Sybil node first enters into the cluster and behave as a normal node initially. Then the Sybil node starts impersonating itself and slowly become the head of the cluster. This makes the illegitimate node at the head of the cluster and thereby disrupts the normal working of the network.

Therefore, Sybil attacks will have a serious effect on the normal operation of wireless ad hoc networks.

5.2 Future Scope

To find other type of attacks which could disrupt the Mobile Ad Hoc Network (MANET).

We can also work on other clustering algorithms like lowest ID clustering algorithms and launch the Sybil attack to disrupt those systems. It would give a deeper insight into MANET and an even better understanding of Sybil attacks.

Since we have already studied how a Sybil attack occurs, it would be great to devise the methods which could defend against the Sybil attacks. Some of the existing methods to defend a Sybil attack are Trusted Certification, Resource Testing, Gate Keeper etc. We can implement these techniques to guard against Sybil attack and even devise a better method to do the same.

REFERENCES

- [1] Aarti and Dr S. S. Tyagi, “Study of MANET: Characteristics, Challenges, Application and Security Attacks”, *International Journal of Advanced Research in Computer Science and Software Engineering*, Volume 3, Issue 5, May 2010
- [2] Mrs Padma and Mr R.Suresh, “Literature Survey on latest research issues in MANET”, *international Journal of Advanced Research in Computer Engineering & Technology (IJARCET)* Volume 2, Issue 7, July 2013
- [3] Alex Hinds, Michael Ngulube, Shaoying Zhu, and Hussain Al-Aqrabi “A Review of Routing Protocols for Mobile Ad-Hoc Networks (MANET)”, *International Journal of Information and Education Technology*, Vol. 3, No. 1, February 2013
- [4] V.Preetha and Dr K.Chitra, “Clustering & Cluster Head Selection Techniques in Mobile Adhoc Networks”, *International Journal of Innovative Research in Computer and Communication Engineering*, Vol. 2, Issue 7, July2014
- [5] Amol Vasudeva and Manu Sood, “SYBIL ATTACK ON LOWEST ID CLUSTERING ALGORITHM IN THE MOBILE AD HOC NETWORK”, *International Journal of Network Security & Its Applications (IJNSA)*, Vol.4, No.5, September 2012
- [6] C. -K. Toh, (2002), “Ad Hoc Mobile Wireless Networks: Protocols and Systems”, *Prentice Hall, PTR*.
- [7] E. M. Royer and C. -K. Toh, (1999), “A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks,” *IEEE Personal Communications*.
- [8] J. R Douceur, (2002), “The Sybil Attack”, *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, pp. 251–260, Springer Verlag, London, UK.
- [9] J. Newsome, E. Shi, D. Song and A. Perrig, (2004), “The Sybil Attack in Sensor Networks: Analysis & Defenses”, *IPSN '04. Proceedings of the 3rd international symposium on Information processing in sensor networks*, pp. 259–268, ACM, Berkeley, California, USA..

APPENDICES

Source Code

```
import java.awt.*;
import java.awt.event.*;
import java.util.Random;
import javax.swing.*;
import java.io.*;
BallWorld b=new BallWorld(xmax,ymax,n);
class BallWorld extends JFrame implements ActionListener {
    JTextField t[]=new JTextField[150];
    public int attack[]=new int[20];
    public int x[]=new int[50];
    public int y[]=new int[50];
    public int dx[]=new int[50];
    public int dy[]=new int[50];
    public int non,atid,atid1,oc1,oc2;
    JTextField tttt[]=new JTextField[100];
    public int xmax,ymax,n;
    JTextField jf[]=new JTextField[100];
    private BallPanel panel;
    JButton b1=new JButton("");
    JButton b2=new JButton("");
    JTextField tt[][]=new JTextField[25][25];
    JLabel l1,l2,l3,l4;
    JTextField mtr[]=new JTextField[20];
    JTextField summ[]=new JTextField[20];
    JButton b3=new JButton("");
    Container c = getContentPane();
    public BallWorld (int xxx,int yyy,int nnn) {
        non=nnn;
        for(int i=0;i<non;i++)
        {msgg[i]=0;
            summ[i]=new JTextField("");
        }
        for(int i=0;i<25;i++)
        bl[i]=100;
        xmax=xxx;
        ymax=yyy;
        n=nnn;
        setTitle("Simulation Area");
        setLayout(null);
        panel = new BallPanel (xmax,ymax);
        int k=0,yy=110;
        l4=new JLabel("NODE PARAMETERS");
        l1=new JLabel("Node id");
        l2=new JLabel("Node x");
        l3=new JLabel("Node y");
        for(int i=0;i<n;i++,yy+=25)
        { t[k]=new JTextField(Integer.toString(i));
            add(t[k++]);
            t[k]=new JTextField("");
            add(t[k++]);
            t[k]=new JTextField("");
            add(t[k++]);
        }
        b1.addActionListener(this);
        b2.addActionListener(this);
        b3.addActionListener(this);
        setSize(1400, 1200);
        setVisible(true);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
    }
    public void actionPerformed(ActionEvent ae)
    {
        if (ae.getSource()==b2)
        { Adjacent1 aj=new Adjacent1(); } }
    class BallPanel extends JPanel implements ActionListener {
        int width, height;
```



```

Ball ball;
private Timer timer = new Timer(300, this);
public void stopt()
{ timer.stop();}
public void startt()
{timer.start();}
public BallPanel (int width, int height) {
this.width = width;
this.height = height;
ball = new Ball();
ball.setBoundingBox( new Rectangle( 0, 0, width, height ) );
timer.start();
}
public void paintComponent( Graphics g ) {
g.setColor( Color.WHITE );
g.fillRect( 0, 0, width, height );
ball.paint( g );
}
public void actionPerformed(ActionEvent e) {
ball.action();
repaint();
}
}
class Ball {
Rectangle defaultBox = new Rectangle(0,0,xmax,ymax);
Rectangle box;
Ball() {
Random r=new Random();
for(int i=0;i<n;i++)
{ x[i]=Math.abs(r.nextInt())%200 ;
y[i]= Math.abs(r.nextInt())%200 ;
dx[i]=-(r.nextInt()%5+10);
dy[i]= -(r.nextInt()%5+10);
}
}
public void setBoundingBox( Rectangle ra ) {
box = ra;
}
public void paint( Graphics g ) {
int k=1,m=2;
for(int i=0;i<20;i++,k+=3,m+=3)
{t[k].setText(Integer.toString(x[i]));
t[m].setText(Integer.toString(y[i]));
}
for(int i=0;i<n;i++)
{g.setColor(Color.GREEN);
g.fillOval( x[i], y[i],20,20 );
g.setColor(Color.DARK_GRAY);
g.drawOval(x[i], y[i],20,20);
FontMetrics fm=g.getFontMetrics();
g.setColor(Color.BLACK);
g.setFont(new Font("TimesRoman",Font.BOLD,12));
if(i>9)
g.drawString(Integer.toString(i),x[i]+5, y[i]+15);
else
g.drawString(Integer.toString(i),x[i]+9, y[i]+15);
}
}
}

class Adjacent1 extends JFrame implements ActionListener {
JButton b=new JButton("START CLUSTERING");
JLabel b1[]=new JLabel[20];
JLabel b2[]=new JLabel[20];
Container c = getContentPane();
public Adjacent1 () {
setTitle("Adjacency Matrix");
setLayout(null);
int yy=80,xx=50;
try
{b.addActionListener(this);
add(b);
b.setIcon(sc);
}
}
}

```

```

for(int i=0;i<n;i++,xx+=50)
{b1[i]=new JLabel("N"+Integer.toString(i));
add(b1[i]);
}
for(int i=0;i<n;i++,yy+=30)
{b2[i]=new JLabel("N"+Integer.toString(i));
add(b2[i]);
}
yy=50;
for(int i=0;i<n;i++)
{yy+=30;
xx=50;
for(int j=0;j<n;j++)
{tt[i][j]=new JTextField("");
add(tt[i][j]);
xx+=50;
}}}
catch(Exception e){}
adjac ad =new adjac();
setSize(1400, 1200);
setVisible(true);
setDefaultCloseOperation(EXIT_ON_CLOSE);
}
public void actionPerformed(ActionEvent ae)
{if(ae.getSource()==b){
BallWorld1 baa=new BallWorld1();
}}
class adjac extends JPanel implements ActionListener {
private Timer timer5 = new Timer(300, this);
public adjac () {
timer5.start();
}
public void paintComponent( Graphics g ) {
for(int i=0;i<n;i++)
{
for(int j=0;j<n;j++)
{
if((Math.sqrt(Math.pow((x[i])-(x[j]), 2)+Math.pow((y[i])-(y[j]), 2))<=120)&&(i!=j))
tt[i][j].setText("1");
else
tt[i][j].setText("0");
}}}
public void actionPerformed(ActionEvent e) {
for(int i=0;i<n;i++)
{for(int j=0;j<n;j++)
{ if((Math.sqrt(Math.pow((x[i])-(x[j]), 2)+Math.pow((y[i])-(y[j]), 2))<=120)&&(i!=j))
tt[i][j].setText("1");
else
tt[i][j].setText("0");
}}}}
}
}
class BallWorld1 extends JFrame implements ActionListener {
JLabel imm[]=new JLabel[50];
JTextField ch[]=new JTextField[50];
int deg[]=new int[50];
int chc=0;
JLabel po=new JLabel("");
public Timer timer2 = new Timer(3000, this);
JLabel l5=new JLabel("CLUSTERS");
JLabel l6=new JLabel("CLuster Id");
JLabel l7=new JLabel("Cluster details");
JLabel l11=new JLabel("Head");
JLabel si,ri,mi;
JTextField ttt[]=new JTextField[50];
private BallPanel1 panel1;
Container c1 = getContentPane();
int cv=-1,cy=110;
int count=0;
BallPanel1 bb;

```

```

JButton sa=new JButton("");
JProgressBar pB[]=new JProgressBar[50];
public BallWorld1 () {
sa.addActionListener(this);
add(sa);
sa.setIcon(saa);
po.setIcon(PHO1);
add(po);
setTitle("Clustering ");
setLayout(null);
panel1 = new BallPanel1 (xmax,ymax);
panel1.setBounds(0, 0,xmax,ymax);
c1.add(panel1);
JLabel l8=new JLabel("Nodes id with degree and battery level");
JLabel l9=new JLabel(" ID");
JLabel l10=new JLabel("Degree");
JLabel l12=new JLabel("Battery Level");
JLabel l13=new JLabel("Message Details");
int yyy=110,k=0,mn;
int yy=110;
for(int i=0;i<n;i++,yyy+=30)
{
ttt[k]=new JTextField(Integer.toString(i));
ttt[k].setBounds(920,yyy,40,20);
ttt[k].setHorizontalAlignment(SwingConstants.CENTER);
add(ttt[k]);
k++;
ttt[k]=new JTextField("");
ttt[k].setBounds(970,yyy,40,20);
ttt[k].setHorizontalAlignment(SwingConstants.CENTER);
add(ttt[k]);
k++;
}
chc=0;
k=0;
yyy=110;
for(int i=0;i<10;i++,yyy+=30)
{ ch[chc]=new JTextField(" ");
cy+=30;
add(ch[chc]);
chc++;
ttt[k]=new JTextField("");

add(ttt[k]);
k++;
ttt[k]=new JTextField("");
add(ttt[k]);
k++;
}
bb=new BallPanel1(500,500);
add(bb);
setSize(2000 , 2000);
setVisible(true);
setDefaultCloseOperation(EXIT_ON_CLOSE);
}
public void actionPerformed(ActionEvent ae)
{if(ae.getSource()==sa)
{ BallWorld2 bbbb=new BallWorld2(Integer.parseInt(ch[0].getText()));} }
class BallPanel1 extends JPanel implements ActionListener {
int width, height;
Ball1 ball1;
public BallPanel1 (int width, int height) {
this.width = width;
this.height = height;
ball1 = new Ball1();
ball1.setBoundingBox( new Rectangle( 0, 0, width, height ) );
timer2.start();
}
public void paintComponent( Graphics g ) {
g.setColor( Color.WHITE );

```

```
g.fillRect( 0, 0, width, height );
ball1.paint( g );
}
public void actionPerformed(ActionEvent e) {
repaint();
}}
class Ball1 {
Rectangle defaultBox = new Rectangle(0,0,xmax,ymax);
Rectangle box;
public void setBoundingBox( Rectangle ra ) {
box = ra;
}
```