Attack Containment in Mobile Ad hoc Networks Through Fair Distribution of Processing Resources

MS Research Dissertation

By

Farzana Azam

(433-FBAS/MSCS/S08)

Supervised By:

Prof. Dr Muhammad Sher

Co-Supervised By:

Mr. Zeeshan Shafi Khan

Department of Computer Science

Faculty of Basic and Applied Sciences,

International Islamic University, Islamabad

2009

1- Mobile communication systems.

2- Wireless „ „

3 - Ad hoc networks (computer networks).

4 - „ „ „ - Security measures

A Dissertation submitted to the

Department of Computer Science

International Islamic University Islamabad

As a partial fulfillment of requirements for the award of

The degree of

MS in Computer Science

International Islamic University, Islamabad

Dated: 23-1-2010

Final Approval

It is certified that we have examined the thesis titled "Attack Containment in Mobile Ad hoc Networks Through fair Distribution of Processing Resource" submitted by Farzana Azam, Registration No: 433-FBAS/MSCS/S08, and found as per standard. In our judgment, this research project is sufficient to warrant it is acceptance by the International Islamic University, Islamabad for the award of MS Degree in Computer Science.

Committee

External Examiner

Dr. Muhammad Zubair
Assistant Professor
Faculty of Computing
Riphah International University
Islamabad

Internal Examiner

Mr. Qaisar Javaid
Assistant Professor
Department of Computer Science
International Islamic University
Islamabad

Supervisor

Prof. Dr. Muhammad Sher
Chairman, Department of Computer Science
International Islamic University
Islamabad.

Co-Supervisor

Mr. Zeeshan Shafi Khan
Lecturer
Faculty of Computing
Riphah International University
Islamabad

Zeeshan Shafi

Dedicated to Shuhdahs of 20th October

Declaration

We hereby declare that this work, neither as a whole nor as a part has been copied out from any source. It is further declared that we have conducted this research and have accomplished this thesis entirely on the basis of our personal efforts and under the sincere guidance of our supervisor Prof. Dr Muhammad Sher and our Co-Supervisor Mr. Zeeshan Shafi Khan. If any part of this project is proved to be copied out from any source or found to be reproduction of some other project, we shall stand by the consequences. No portion of the work presented in his dissertation has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

Farzana Azam

Farzana Azam

(433-FBAS/MSCS/S08)

Acknowledgement

All praise to Almighty Allah who has all the names, and who need no name the most generous, considerate, and compassionate who has blessed mankind with this verdict to think, explore, to learn and discover the hidden secrets of this universe and helped me to broaden the veils of my thought and enabling me to get through the difficulties indulged during this project. Also admiration to our beloved Prophet Muhammad (PBUH) who is always a great source of inspiration of divine devotion and dedication to me.

I would cordially pay my special appreciations and whole heartedly considerations to my reverend supervisors Prof. Dr Muhammad Sher and Co-Supervisor Mr. Zeeshan Shafi Khan for their endless support, guidance and coordination while conducting this project. I owe them a great respect and honor and I am privileged to work under their supervision. It was their efforts, courage, moral support and endeavoring attitude that helped me to get through any problem or difficulty during each step of this project.

I would also like to pay my gratitude to all my respected teachers making me capable of what I am today due to their guidance and help. Thanking Miss Muneera Bano for her views which helped me in improving the proposal, also Miss Zakia jalil and Miss Nazli for not only their moral support but also providing the managerial and technical support as well.

Thanking my friends for always being there for me whenever I needed them for their help, generosity and moral support.

I also owe a special thanks and gratitude to Dr. Neil Daswani and Stanford University for supporting me in working on this idea and accompanying me with the material for study which I needed during this research.

Finally my beloved parents and family who deserve the credit more than I could ever express for always being completely supportive to me. They have been a constant source of advice, Love and devotion to me. From moral to financial they have been blessing me with all the support that I have needed up till now in my life.

I express my countless appreciation to all the people who have helped me during achieving this MS degree and hope to have this honor that they would walk along me through out my life.

Farzana Azam

Project in Brief

Project Title:	**Attack Containment in Mobile Ad hoc Networks Through fair Distribution of Processing Resource**
Undertaken By:	**Farzana Azam**
Supervised By:	**Prof. Dr Muhammad Sher**
Co-Supervised By:	**Mr. Zeeshan Shafi Khan**
Start Date:	**January 2009**
Completion Date:	**December 2009**
Tools and technologies:	**Network Simulator 2**
Documentation Tools:	**MS word, EDraw, MS Excel**
Operating System:	**MS Windows XP professional**
System used:	**Pentium 4, 1.73 GHz**

Abbreviation Used

Abbreviations	**Acronyms**
P2P	Peer-to-peer
MANETs	Mobile Ad hoc Networks
DoS	Denial-of-Service
SSS	Security Attack, Security Mechanism and Security Service
NS-2	Network Simulator-2
Q	Capacity of Processing Queries at a time
QDS	Query Distribution Strategies
T	Number of Trusted Nodes

Abstract

Ad hoc network is a network that does not rely on any infrastructure and created and maintained by individuals to share information with each other. Security of this infrastructure less network against different types of attacks is one of the alarming issues. To obtain and share information with each other, nodes of the ad hoc network use the flooding algorithms. A single query transmitted by a node can be received by hundreds of nodes at 3rd or 4th hop. Use of these flooding algorithms provides an opportunity to an attacker for launching a query flooding attacks. On one side these query flooding attacks results in wastage of valuable processing resources and on the other side results in starvation and delay at legitimate user's end. Currently ad hoc network does not have any mechanism to minimize the effect of an attack during the attack time. To solve this problem we proposed attack containment techniques by providing a fair share of processing resources to every node. To achieve the fair distribution we proposed different query distribution strategies which allocate the resources on the basis of specific mathematical models. The validation and performance of these query distribution strategies will be evaluated by using NS-2.

Table of Contents

List of Figures

1. Introduction

1. Introduction:

Earlier before the advent of computer there was a need to protect and secure the information and data which was valuable for an organization. Any administrative or physical means were used to keep valuable information confidential. For example earlier such types of confidential files were kept under locked cabinets. So we see that the concept of security is not new.

Although this concept has flourished very well with the advent of computer. We need automated tools to secure all the files and data which are stored in a computer. This is needed incase where we have a shared system or especially when we want to access it over some public telephonic network or internet. The general term under which this phenomenon of protection comes incase of computers is known as computer security.

The term security has rapidly grown with the use of networks. Networks are build to provide communication between computer and the end user, also between computer to computer. So there is a need to make the network secure as much as one can, so that the data even remains secure during its transmission. To protect data over a network we require special security measures which in networks come under the term of network security. With the vast growing networks today this term is being improved day by day. Now the security measures are made which are according to the need of a particular network's requirements.

We see many different networks that are currently being used like we have the traditional client- server based networks, the widely used data sharing networks also known as peer-to-peer(P2P) networks, and the most emerging networks of today is Mobile Ad hoc networks (MANETs). All of these networks are victims of different types of attacks. Of these attacks the denial-of service (DoS) attacks badly effect the performance of these networks. As securing a network is becoming the most important need of today of all types of networks, so let us study and compare the working and performance of these networks under such attacks. Details are as follows.

In the traditional client-server based network the Internet the servers are the victims of various kinds of attacks. Servers are vulnerable to them. If due to any malicious attack some of the servers are made disabled from performing their tasks, then many of the legitimate clients might become unable to continue their work or continue their functions. For example if we look at some big sites like yahoo, Amazon, eBay, hotmail or search engine like Google, Alta Vista etc all of them are based on client-server model all of them have been victims of denial-of-service attacks(Dos) in past few years and yet they are vulnerable to these attacks. The results of these attacks even lasted for the whole day i.e. the servers were unable to recover for the whole day.

In these attacks an intended attacker gets commands over some of the clients of the network. Then he instructs those clients to generate large amount of data and send them to be processed by the servers. As the servers are few, so they very soon become overwhelmed by this large amount of data. Due to which servers are unable to respond to the requests from their legal clients. This results in complete network resources loss. This great loss of resources is due to the client-server model of Internet. Where few servers are designated to provide services to a huge number of clients, which provide an added opportunity to an attacker to make such attacks easily and frequently and easily contain the whole network.

Where as in the case of peer-to-peer (or P2P) networks there is no discrimination among the peers (nodes) i.e. any of the nodes can work both as a client or a server simultaneously. All nodes can perform the functions of a server. They work in a distributed environment with cooperation, and hence share the network resources like bandwidth and storage capacities. Like if any request is made by a client (user) to any one of the node in the network, if that node does not have the requested data, it will send that request to the next node and so on this continues till the requested data is found and delivered to the recipient. In a P2P network nodes rely on each other to perform any task, especially during any application layer task such as searching documents. In such systems nodes share the work load of each other nodes and work on each other behalf's.

If any part of the network is affected (due to any denial-of-service attack) yet the clients are able to carry out their tasks due to the reason that the functionality of an individual server is being distributed among different peers. So if any of the requested node is unable to deliver the required services, any other active node in the network can provide those services and fulfills the request of the clients on behalf of that victim node. Which is being affected by the malicious activity. Though P2P networks are vulnerable to application layer DoS attacks but due to the server's distributed functionality among various nodes the malicious effects can be reduced which allow clients to continue their tasks.

In contrast to these networks Mobile Ad hoc networks (MANETs) perform differently. It has no specified fixed network architecture as other networks have this is because all the nodes which are part of this network are mobile. There is no discrimination among the nodes as server or client, all the nodes can send and receive data simultaneously. The devices or nodes become the part of this network for a small duration and they are allowed to dissipate the traffic to any of the node with in the network at there will. All the devices are mobile, they can move in any direction this nature of nodes make this network spontaneous and self configuring.

The MANETs are one of today's emerging networks. They use flooding based protocols for disseminating data or message with in the network. When a node is to transmit message to any off the node, due to the flooding nature of routing protocols this message will be routed to all the nodes within the network. So this flooding nature of protocols provides an added opportunity to the attacker to commit an attack very easily and get hold over the entire network resources with in very short period of time.

Lots of work is being done in securing this network against different types of DoS attacks. Mostly the work is done over network layer i.e. making MANETs resilient to network layer DoS attacks. But the application DoS attacks appear to be more damaging then the network layer attacks. Where a single malicious node can overwhelm the entire network by generating few queries. Because due to the flooding based protocols a single query can be received at more than hundred nodes in third or fourth hop. This would result in not only wastage of network resources but will also deny the legitimate queries to be processed. Hence the legitimate clients or users will starve to get their network share. This attack may become more and more injurious as the number of the malicious nodes increases in a network and would leave the network to remain in-active and non-functional for the whole day even. This is due to the reason that currently MANETs don't have any mechanism to mitigate these malicious effects during the attack time.

If we see at a glance the techniques that have been developed till date to deal with denial of service, their main context is network layer DoS attacks. We see no such preventive techniques regarding application layer attacks. Whenever an intended attacker gets entry in to the network it tries to maximize its effect by getting hold over the entire network resources. It starts broadcasting its queries over the network, then all the legitimate nodes will also be indulge in processing these malicious queries. This act will not only occupy the valuable network resources like its bandwidth but will also occupy the individual nodes processing capacities and their other valuable resources. Thus making the nodes busy and unavailable for the legitimate users to place their request. Hence this would result in complete network loss and in very short time the network would become disabled to facilitate its legitimate users or clients. So there is a need of such techniques that can give each node a fair share of processing capacity with in the network.

This massive malicious act rapidly grows in the network because there is no mechanism in the MANETs to restrain this kind of act during the attack time. They lack such techniques to minimize the effects of illegal acts. Due to which malicious effects can not be controlled and hence unable to protect the whole network from malfunctioning and disabling. Thus the need of fairly distributing processing capacity and attack containing techniques to minimize the chances of starvation and preventing the entire network from disabling is the main focus of this research presented in this thesis.

In our dissertation we formulate and evaluate such mitigating techniques that will minimize the application layer DoS attacks in MANETs and also provide fair distribution of resources. For this fair distribution and containment, we propose different query distribution techniques which are based on mathematical models. Thus we have enhanced security measures in MANETs by introducing attack containment techniques.

This research has also justified the use of flooding algorithms in MANETs. The validation and performance of these query distribution techniques has been evaluated by using NS-2. All these techniques are formulated while considering the battery power and energy consumption issue. Regarding this all the techniques are cost effective because they make maximum tries to consume less energy while working.

1.1 About Mobile Ad Hoc Networks (MANETs):

A temporary network which is usually created for a small period of time is known as Mobile Ad hoc Network (MANET). They are also known as Mobile Mesh Networks. It is one of the types of wireless ad hoc decentralized networks. It is composed of all mobile and loosely coupled nodes. They can move in any arbitrary direction, so they are free to become part of any network at any time they want to. Hence it is a dynamic topology based self configuring network. This is basically a user oriented network i.e. individuals formulate and maintain this type of network to share information with each other [2].

The protocols which are used to route the traffic in MANETs are based on flooding algorithms i.e. a single message is broadcasted over the entire network this seem to be wastage of network resources but this attribute adds the feature of reliability of a message to this wireless link network.

It does not have any physical back bone rather it has a distinctive, particular node called as access point that manages communication with in this infrastructure less network. On top of Link Layer ad hoc network MANETs implement a routable networking environment. Some salient features associated with MANETs which are as follows.

- ✓ Dynamic topologies
- ✓ Bandwidth-constrained

- ✓ Energy-constrained operation and
- ✓ Limited physical security
- ✓ No dedicated network components

MANETs can be found both in standalone state and also in extended form by connecting it to the Internet. The major challenge which the nodes of this network faces due to the randomly changing network topologies is that they must be continuously properly equipped with the information which is required traffic routing. Mobility makes difficulties for wireless communication to establish and maintain relationships among nodes or users.

They are mostly used to form future home or an office networks. Where we can quickly add or remove a device from the network easily. We can use the emerging wireless technologies like Bluetooth, infrared transmission or radio frequency and jini to form an ad hoc network. Bluetooth allows wireless communication among mobile users, computer and other devices. Infrared transmission or radio frequency allows conferees to come across each other by connecting there notebook computers to local network. Similarly jini is able to detect new devices in a network instantly. Thus all these technologies mentioned above provide an easier and quicker way to form an ad hoc network at a user's will [2].

Securing this type of network which is fragile, whose links are wireless with dynamic topologies having limited bandwidth and energy is itself a great challenge to work on. Hence it requires such security measures which should be developed while considering these issues.

In many different ways MANETs are different from the other networks. The differences are as follows.

- ✓ They have a decentralize network. Where all the tasks of network is done by the nodes themselves. As the nodes incorporate the function of maintaining the routing information. Therefore the task of discovering topologies and delivery of messages is performed by them.

- ✓ All the nodes or devices are mobile and they use half-duplex mode of transmission.
- ✓ They support mobility to some extend.
- ✓ Network is not created by network, instead links or nodes form the network.

1.1.2 Types of Mobile Ad Hoc Networks (MANETs):

Types of MANETs along with a brief introduction of each type i.e. where and when these networks are formed along with there utilization is explained as below [2].

- **Vehicular Ad Hoc Networks (VANETs):**
  As the name indicates it is used by the vehicles to communicate with each other and also to provide communication between vehicles and other equipments at the road.
- **Intelligent Vehicular Ad Hoc Networks (InVANETs):**
  It has an added intelligence to it and provides different automatic intelligent solutions, which enables the vehicles to behave intelligently during any unpredictable situations like accidents, during drunker driving, collision among vehicles and other erratic circumstances.
- **Internet Based Mobile Ad Hoc Networks (iMANET):**
  These Ad hoc networks are used to provide connection among portable nodes and the internet-gateway nodes which are fixed.

1.1.3 Applications of Mobile Ad Hoc Networks (MANETs):

This decentralize, wireless and dynamic nature of ad hoc networks make them appropriate to be used in such applications where one can not rely on the central nodes.

As MANETs require very less configuration, which makes their deployment very easy. So they become the best suitable networks that are needed to be formed in the situations of emergencies like war, natural disasters (earth quakes, floods etc).

The added feature of dynamic topologies and the adjustable, adaptive nature of its communication protocols make this network to be constructed quickly with less consumption of time than a wired network would be taking.

On the bases of applications the Ad hoc networks can be classified in to following types

- ✓ Mobile Ad hoc Networks (MANETs)
- ✓ Wireless mesh Networks
- ✓ Wireless Sensor Networks

In this dissertation we have studied only Mobile Ad hoc networks (MANETs). Its security measures are enhanced here by introducing the concept of containment in MANETs.

1.1.4 Advantages of MANETs:

MANETs are getting popularity day by day due to some main advantages like convenience, ease of deployment, cost efficiency and other benefits associated with them. These abilities make them more efficient and randomly progressing networks of today. Which are described as follows [2].

- **Mobility:**

  Due to the wireless nature of these networks, users can even have an access not only to this network but also to the Internet from any location he wants to other than its working environment. Users have any of these wireless technologies like Bluetooth, Wi-Fi, infra red built-in in their devices which support and provide them this mobile connection to the Internet.

- **Deployment:**

  While setting up this infrastructure less network it requires just few configurations, but as compare to it while setting a fixed wired network it requires more configurations as well as it adds other complexities of cost and difficulties involved in while handling this physical media (cables) running across various locations.

- **Scalability:**

  Wired networks always require additional cables and equipments whenever the nodes are added to the network. But these networks with the same equipments can support and serve the sudden addition of clients or nodes. These networks are not effected by the random joining or leaving of nodes.

- **Cost:**

  These wireless networks are becoming more cost effective then the wired ones because as they have eliminated the use of physical cables. Thus they have saved the cost of purchasing and labor involved in handling them.

- **Convenience:**

  As these networks support mobility. Thus this feature provides its users to have an easy access to the network resources from any convenient location with in their office or home.

- **Productivity:**

  Users of wireless networks seem to be more productive then those of wired ones. That is because the users of MANETs can nearly maintain a continuous connection to their respective network while moving from one place to another. So an employee can nearly be all times available to its organization and thus will be more productive as his work can be completed from any convenient place.

1.1.5 Disadvantages of MANETs:

We have seen some of the benefits of MANETs as above. They also have some downfalls. There are some reasons due to which these networks are not preferable in some situations. This has to do with some built in limitations which are associated with this wireless technology. Those limitations are explained as below [2].

- **Reliability:**

  There is no dedicated media to rely on in case of MANETs. Messages are sent over wireless links in the form of signals by the mobile devices. These mobile signals are always subjected to a variety of interruption while travelling. Like for example mobile signals are always at a threat of interference by the microwaves. The performance of MANETs is badly affected by these types of waves, which put a great impact on stability and reliability of MANETs. This is another reason of using MANETs to make small area networks.
- **Bandwidth:**

  MANETs have low capacity links. The mobile users interact with this wireless network through this bandwidth. This is constrained here.
- **Range:**

  All wireless networks have some specified fixed range in which it is reachable. Similarly in the same way MANETs also have. That is why they are more feasible to make small networks then to make networks with large infrastructure.
- **Radio Emissions:**

  As the medium used by these networks is wireless. They use wireless technologies like Bluetooth, infra red etc for communication. All these technologies use radio frequencies for propagation of a message from one end to another. These emissions upon interference with any other signal or device might become injurious to human health.
- **Security:**

  Security of such a wireless link networks is itself a big challenge. MANETs do not have any infrastructure nor any dedicated media to rely. The message is subjected to an open medium i.e. air. Therefore it requires very strong encryption techniques to protect its message from interruption.

- **Energy and Power consumption:**

  As all the devices in MANETs are Mobile. To remain alive all of them require some kind of energy or battery power. I.e. periodically they require some kind of charging, without this periodic charging the devices or users will not be able to have a continuous live connection to the network.

1.2 Technical Detail of MANETs

As we see the definition of MANETS, we come to this point that MANETS have no specific or particular architecture. Because it supports mobility, so its nodes are free to be part of any MANET at any time. Due to this mobility there is no proper infrastructure of MANETs. This is an added feature of this network.

One node in one session can be a part of one MANET and in the very next session it can be the part of another MANET. As the media is wireless due to which every node can dynamically and easily join and leave a network and change its physical location. So due to this dynamic nature we are unable to forecast the position of a single node. Mobility enables each node to form a dynamic network around itself. Which also indicate that the neighbors of each node also change over a period of time. Predicting a node's neighborhood also becomes difficult. This dynamic, mobile nature of nodes allows the nodes to freely move in any direction with in a network. In some scenarios it is some times even impossible to predict a MANETs' membership due to its spontaneous, mobile, and wireless nature [3].

Each MANET has a specific range or area with in which a node can be accessed or reached. With the help of figures it has been shown that MANETS lack a particular fixed architecture.

They have no physical backbone nor require any physical media to rely on for communication among nodes. All of the nodes are connected wirelessly to each other.

Diagrammatically this infrastructure less network is shown as below. Also showing the division and re-mergence of one MANET to another.

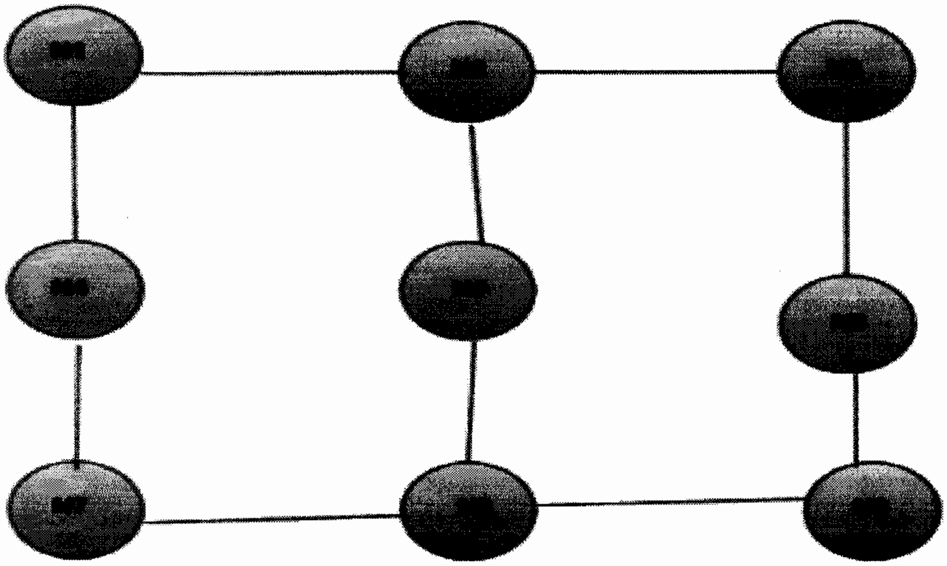

Figure 1.1: MANET containing set of nodes

In this figure we see that at this moment there are nine nodes, which are part of this MANET. But if we observe this MANET for a while we may see that this single MANET would divide in to other MANETS with the passage of time.

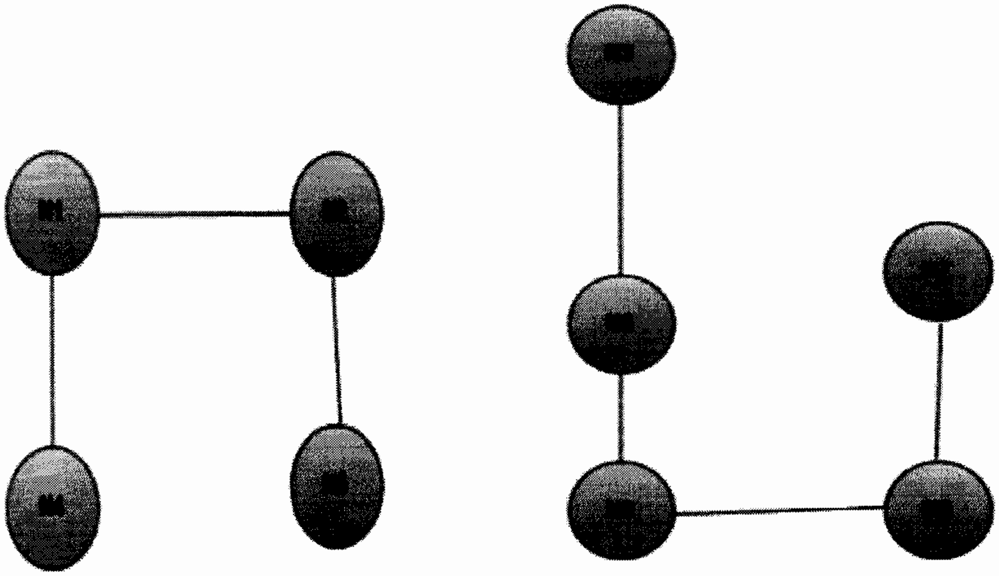

Figure 1.2: with passage of time One MANET has divided in to Two MANETs

With the passage of time or in the very next session this MANET could partition in to two more MANETs as shown above. Later on these MANETs can even remerge back to form one large MANET. Even these nodes can merge to form many different MANETs [3].

The reason behind these changes is the nodes mobility. MANETs do not bound the nodes to remain fixed to a single network. Nodes can change there position at any time. They can join or leave a network at their will. Due to this property we see that a node's neighborhood also changes frequently. In MANETs flooding based protocols are used for communication, this is best suitable for them as they have loosely connected nodes. Through flooding a packet will be disseminated to all the connected nodes of the network. Due to this nature securing such infrastructure less network is challenging.

There is no standard classification of ad-hoc networks. But a simple classification is to distinguish by formation and communication. In Single- hop nodes are in their reach area and can communicate directly where as in multi-hop nodes are far away and cannot communicate directly. So they communicate with each other through intermediate nodes.

1.3 Flooding Algorithms:

In flooding algorithms the message or packet is disseminated to each node within the Network. Flooding based communication strategy enables the message to be delivered to all active nodes of the network.

In flooding algorithms nodes behave as follows [2]:

- Every node can behave as a transmitter or a receiver.
- Every node forwards the message to its neighboring nodes, other than the source node.
- Due to the broadcast nature of flooding algorithms certain challenges are associated with flooding based protocols.

- One message with a single destination has to be traversed over the entire network. This results in increased network load and wastage of network bandwidth.
- Duplication of messages can also occur which would generate excessive load on the network since we require additional complex processing techniques to avoid these duplications.
- Selective flooding is a technique which can partially address these issues of flooding based protocols.

The advantage that lies in using flooding based protocols is increased reliability. It ensures that at least once the message will be delivered to each node of the network. It almost guarantees the source to destination delivery of the message.

1.4 Use of flooding Algorithms:

Flooding algorithms are widely used in systems such as Usenet, peer-to-peer file sharing systems, in some routing protocols like OSPF, DVMRP and in protocols that are used in Mobile ad hoc networks. To share information with each other nodes of MANETs mostly use flooding algorithms [2].

There are many reasons to choose flooding based protocols for MANETs.

- Of them the most important reason is, that the nodes in a MANET are mobile. Due to their mobility they too often join and leave the network. This raises the issue of reliability. In this situation flooding based protocols are most supportive because they are very resilient to nodes joining and leaving and hence provide increased reliability by guarantying the message delivery to each node at least once.
- These types of protocols are simple due to which they are also easy to develop and implement in a network.

- Results can also be collected faster while using these flooding based protocols.
- They have better performance and are also used in many real systems.

1.5 Network Security:

A network is created in order to facilitate individuals, by providing them a single platform through which they can communicate with each other and also interact with other networks and their users. A network is composed of some valuable resources which always need to be protected. For this reason we require network security. Network security consists of such measures to protect the network and valued accessible network resources against any un authorize access. It also consistently monitors and measures the networks' performance, efficiency and its effectiveness.

One of the most important question that came into existence with the emergence of the concept of network is, what if a network is not secure, if there are no security measures over the network. Then there would be loss in not only privacy but also confidentiality, integrity, and availability of resources. This would result in wastage of network resources and starvation at the legitimate users end. Because due to the un authorize access the illegal node enters the network and occupies the network resources, due to which the legitimate node fails to get its share and starves. So in order to save these resources and let only the legitimate users or nodes to have an access to the network we use different security policies to protect the network against any kind of threats or attacks [4].

The different policies which are used for protection allow the network to remain functional and perform its desired operations in the adverse conditions i.e. whenever the network is attacked by any illegal access or any other malicious activity, hence from all those activities which cause hindrance in performance of the network.

Networks need appropriate security policies that are to be implemented to allow the network to achieve maximum reliability, performance and efficiency. So we need techniques that can prevent, detect, contain and recover a network from any kind of malicious activity. When and how these policies work and which is best suitable under which circumstances, is briefly explained as follows.

1.5.1 Attack Prevention:

It is one of the defensive measures. Prevention techniques as the name indicates make the attempts of an attack on a network to be unsuccessful to cause any damage. These types of techniques allow the network to hinder any kind of malicious activity or attack. These techniques are developed and implemented in the networks in such a manner that they enable the networks to restrain the attempts of an attacker to commence an attack, and hence securing the networks from any damage, and causing the failure of attack.

Prevention techniques make the network so strong that it is capable of opposing the attempts of an attack, and not allowing it to cause any damage to the network and its resource. Infiltration is an attack in which a malicious node tries to break- in to the network and occupies one or more available resources of the network to its own use and will. Here a preventive technique Firewall can be used. Thus this technique not only prevents the entry of malicious nodes and other adversaries from joining the network, but also opposes the attempts of many DoS attacks as well.

Another way through which malicious nodes try to enter the network is by “Lying”. That is these malicious nodes pretend to be the legitimate nodes of the network and hence make there way to the network. One of the preventive technique which is Digital Signature is used. This technique prevents the illegal node from pretending to be legal by comparing its signature with the legal ones. Therefore it prevents the nodes from lying and secures the network [4].

At a glance we look at all the preventive techniques that are available along with their functionalities i.e. which technique is applicable in what situation. For example in order to control lies one can use Time Stamping, Non Repudiation, Digital Signature, Authorization, Access Control and Authentication. All these techniques protect the networks against the false claiming of any malicious nodes [4].

Similarly the Firewalls, Anti Viruses, Cryptography, and Intrusion prevention system prevent the networks against any Infiltration. Thus by disallowing the illegal, malicious nodes entry in to the network, also preventing the network against any other malicious activities like hacking and many DoS attacks [4].

Prevention techniques make all possible ways to protect the network against any kind of attack or threat that can be made. They are seen and found successful in this regard. But however in certain circumstances like when heavy burst of bogus queries are made to commit DoS attacks against the availability of a network, this approach fails and the malicious attempt make their way to the network.

1.5.2 Attack Detection:

It is to detect or find out the attack once it has occurred. Once the attack has happened, measures are made to identify the attack in order to find out all the resources that are being misused by the malicious activity. Hence recover those resources back and allow the legitimate activities to continue their respective tasks.

Trades off involved in implementation of access control of networks like for Peer-to-Peer and MANETs are challenging. Because if we make strong and complex restriction policies on the type of data which these networks share, this would limit the utility of these networks. Now these networks can not be left un- restricted, otherwise they would become a platform where any type of activity either legal or even illegal can freely distribute any content over the network. Thus the mid way among this is to have sensitive and effective detective techniques. Which will not affect the networks utility, and if any damage or problem is caused to the network it will be detected and the network would be retained to its original state.

Detection also detects those malicious nodes that have entered the network. It not only identifies them but also recovers back all the valuable resources of the legitimate nodes from them. These techniques also provide the awareness about the attack and the damage caused by it to the administrator of the network. So that making him able to take suitable security measure according to the current scenario.

Some of the detection techniques are mentioned here. In an Intrusion Detection System, whenever the malicious nodes try to enter a network, this technique is quickly revoked and it identifies these illegal nodes and stops them from entering [4].

Similarly the Quantum System works, whenever any intruder tries to break the encryption key, this technique is revoked, which identifies and quantifies this malicious act [4].

In case of node or any other resource failure the detection techniques like Watchdog Processors, Polling and Beacons are used to identify these types of failure and recover the network back in to its original state [4].

Some of the lies which dodge the prevention techniques and get entry to the network. Then the detection techniques like fail-Stop Digital Signatures are used to identify these lies and restore the network back from the effects of these lies [4].

When any of the infiltration escapes prevention then the detective techniques like Tripwire and Viruses Scanners can be used by the network to detect infiltration and to recover the damages made by it [4].

However as to every network there is a threat. Similarly in the same way these detection techniques are susceptible to many of the DoS attacks. Those can make their way through these techniques.

1.5.3 Attack Containment:

In attack containment measures are made to minimize the effects of an attack. Once the attack has been detected, then techniques are used to control the damage caused by the attack. The main goal of containment is to reduce the effects of the attack such that the network is able to remain functional in the light of the attack.

In some critical situations when a very strong attack is being commenced which goes un-detected and prevented and hence starts overwhelming the network by its actions, at that time there is a need of these containment techniques. Because even during the attack these techniques are functional. That is as the attack is dispersing its effects in the network, these techniques contain its effects, thus reducing the malicious effects while keeping the network operating. Containment performs both of these tasks simultaneously side by side.

Whenever we are unable to remove an attack then we take all possible measures to cater down its effects. In the absence of these techniques a single malicious node can get hold over the entire network by capturing all of the valuable resources of not only the network but also the resources of the individual nodes. Thus the network would be unable to carry out its desired function, and also becomes unable to serve its legitimate users. A complete starvation would be prevailing at the legal users end.

With the help of these techniques we can prevent the whole network from getting down. These techniques work in such a fashion that they discourages the malicious nodes effects as they are being dispersed by containing them and preserve the network and its resources to be available to the legitimate users and back-offs starvation, and network remains functional with desired out put.

One of the containment method is Replication. In replication we make copies of the required data or files and store it at multiple locations, i.e. instead of placing the required data on any one node we store on multiple nodes of the network. Now if due to some malicious activity the node with the actual file crashes the request of the user can be fulfilled by retrieving the data from any one of the storage space where replicated data is being stored [4].

If the network is unable to prevent the lies, then we can make the network capable of tolerating lies and yet allow the network to perform correctly even under the effects of lies. In order to contain these lies the networks can use Byzantine Agreement Protocols and Reputation Systems [4].

In order to contain Infiltration and let the network perform its desired tasks while infiltration is active, networks can use techniques like Virus Cleaners and Intrusion Tolerance.

Techniques like Replication and Backups which can be used to contain the effects of any storage or process failure both work in similar fashion. Other techniques like fault-tolerance, Non-Stop processors and RAID (Redundant Array of Independent Disks) can be used to contain such failure [4].

1.5.4 Attack Recovery:

Attack recovery is to get rid of the damaged caused by the attacks and to restore the network back to its original position. Once the attack is detected then maximum tries are made to confine its effects, then such techniques are used to repair the resources that were being damaged by the malicious activity or attack. Efforts are made to restore the network to its original state in which it was before the attack. So that legitimate work can be done by the network for which it was made.

Recovery techniques restore the networks from under going damage caused by the malicious activities. Recovery makes all those resources which were occupied by the attack available back to the legitimate users, hence putting the network back in to its original state, a state in which it was when constructed. So to resume back its original activities or tasks to out put its desired results for which it was designed.

If due to any failure some of the data or information is lost, with the help of recovery techniques the lost information can be restored back. Like for example if a user encrypts his some of the files with any of the private key and stores that key on to any of the storage media like floppy disk or hard disk. Now due to any failure the storage media fails. This would make nearly impossible to decrypt the files.

By taking the advantage or recovery technique like Escrow we can get rid out from this situation. This will retrieve the private key along with any other encrypted information with it [4].

Other techniques that can be used to recover from any failure of storage or processes are Rebooting or Restarting, Hot Swapping and Fail-over [4].

In order to recover a network from the effects of Lies (pretending of malicious nodes to be legitimate), a recovery technique Auditing is used. A system can recover from all the damage caused by lies when the system under goes the process of Auditing [4].

Another technique that recovers the system from the damages caused by Infiltration is Certificate Revocation i.e. cancelling the older certificates (which might have been hacked and used by intruders to get entry to the system) and allocating the new certificates to the nodes to recover the system or network from Infiltration [4].

1.6 The Other Standard Security Architecture:

In this section we have presented some of the related research of security architecture which is required by nearly every organization to secure its networks from illegal access. Here a systematic approach is used to define the security requirements of a network then developing approaches to satisfy these requirements. It starts from security attacks i.e. showing the types of attacks that can effect the performance of a network, then how and which mechanisms to choose, so to use services to counter the attempted attack. The need of discussing this frame work here is that this supports our work by enabling us to develop efficient security measures while considering the aspects on which this frame work lies. The frame work has three main aspects on which it focuses which are Security attack, Security mechanism and Security service. We represent them as SSS here. First 'S' stands for Security attacks, second 'S' for Security mechanism and the third 'S' for Security service These terms are defined as follows [1].

- **Security attacks:**

  It is an illegal intelligent act with an intention to escape security services and to breach the security policies of a system or network.
- **Security mechanism:**

  A method developed for detection, prevention and recovery of a system from an attack.
- **Security service**

  These services are made with the intentions of to counter act against security attacks. These services can use more than one security mechanisms to protective techniques, which can provide security to the system or network.

After this brief introduction about the aspects or components of the mentioned security frame work or architecture, now the detailed explanation about each concept is presented here.

1.6.1 Security Attacks:

Security attacks have been classified in to two main types which are Active Attacks and Passive Attacks. An active attack makes efforts to captures system or network resources and puts impact on the functions of the system by altering them.

Where as passive attacks only hacks the information from the network without affecting its resources. Active attacks are more injurious then the passive attacks [1].

Passive Attacks:

Passive attacks are an attempt to snoop or monitor the transmission of information. Here the target of the attacker is to gain the information which is being transmitted. The two types of passive attacks are as follows.

- **Release of message contents:**

  In this attack the goal of the opponent is to learn the confidential information transmitted via a telephonic conversation, through an electronic mail message or it may be present in a transferred file. We make efforts to prevent this illegal learning.

- **Traffic analysis:**

  If we prevent the contents of a message from illegal learning, by using any masking technique like encryption, so that if contents of the message or any other traffic is being read but the opponent is unable to extract any information from the message. But yet the opponent can observe the pattern of messages and can determine the position and identity of the communicating nodes, the size of the messages their frequency with which they are being transmitted. All this information is useful in determining the nature of communication being carried out.

In these attacks emphasis is on prevention than detection because these attacks are very difficult to detect. Because they don't cause any disturbance to the transmission between the source and destination neither make any changes to the transmission among them nor any alteration to the traffic. The transmission is being carried out in normal fashion. Hence these attacks hide and just sniff the traffic and they are not visible to any one, so to detect these hidden third parties is difficult. Efforts are made to prevent them. A scenario of passive attack's traffic analysis is shown in the blow figure.

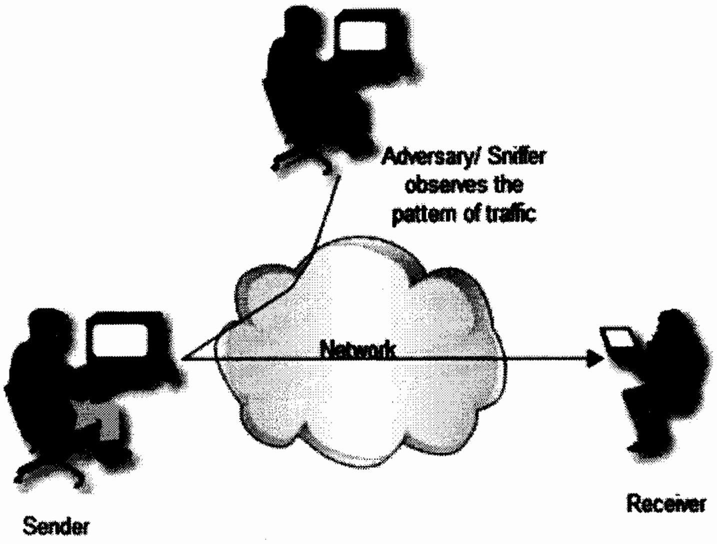

Figure 1.3 Traffic Analysis

Active Attacks:

These attacks make some alteration to the existing stream of data and even they are also responsible of creating fake data stream. These attacks can be categorized in to four types Masquerade, Replay, Modification of messages and Denial of service. Each of the category is explained as below [1].

- **Masquerade:**

  This attack takes place when one node pretends to be any other node. Thus taking the advantage of accessing those privileges that are authorized to the other node, i.e. one entity takes the identity of other entity.

- **Replay:**

  In this attack the attacker captures the on going message and then reproduces it and retransmits it to cause malicious effects.

- **Modification of messages:**

  In this attack some part of a legal message is altered, or the series of the transmitted messages are reordered and delayed to generate a malicious effect.

- **Denial of service:**

In this attack the intended attacker tries to degrade the performance of the network or system by overloading it with the un-necessary traffic which is beyond the capacity of the network it can handle. These attacks are the main focus of this research.

Active attacks are difficult to be prevented because the wide varieties of hardware and software resources are vulnerable. So these attacks are usually detected to recover a system or network from any damage they have caused. Where as passive attacks are prevented because they can not be detected. Figure shows one type of scenario.

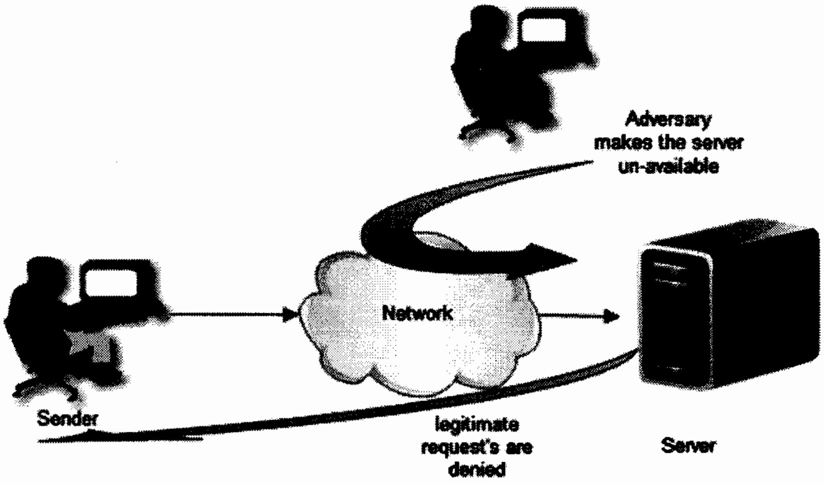
Adversary makes the server un-available
Network
Sender
legitimate request's are denied
Server

Figure 1.4: Denial-of-Service

1.6.2 Security Services:

It is defined as a service provided by the system or the network to its resources so that they can be protected against any kind of malicious activity. Security policies are implemented by security services and security mechanisms are used to deploy these policies. The services are divided in to five main categories and each category is further divided in to sub-categories. These are explained as below [1].

- **Authentication:**

  It is to assure that any user is what it claims to be. It allocates every user of the network an Id and a password through its confirmation only the user is allowed to access the network. This authentication is always carried out when ever a user attempts to access the network.

  It has further two types one is Peer Entity Authentication and the second one is Data Origin Authentication.

- **Access Control:**

  Through access control the use of resources of the system or network is protected against any unauthorized access. The service controls that which user has the privilege to which resource and to which not.

- **Data Confidentiality:**

  It is to protect data against any illegal access. Its types are named as follows. Connection Confidentiality, Connectionless Confidentiality, Selective Field confidentiality, Traffic Flow Confidentiality.

- **Data Integrity:**

  This service ensures that the data received at the destination end is exactly the same, which was sent by the legitimate sender. That is the data is received in its original form and is not being altered or replayed.

  The further types of Integrity are Connection Integrity with Recovery, Connection Integrity without Recovery, Selective Field Connection Integrity, Connectionless Integrity and Selective Field Connectionless Integrity.

- **Non Repudiation:**

  Provides protection against the denial of service. In this attack a single node takes hold of the network and participates in the activities (communication) that are being carried out by the network.

  Its types are one Non repudiation Origin and the second is Non repudiation Destination.

1.6.3 Security Mechanisms:

A specific method used by the system or Network to either prevent, detect, or recover the system back from the damage caused by the attack. It is categorized in to two main types some mechanisms are employed in the specific protocol layer known as specific security mechanisms and some are not specific to any fixed protocol layer known as pervasive security mechanisms [1].

Specific Security Mechanisms:

These may be deployed to any suitable protocol layer so to provide some security services.

Some of its types are briefed here.

- **Encipherment:**

  In it mathematical formulas or algorithms are used to convert the plain text in to cipher text (un-readable form) with the help of encryption techniques so to protect the data against any alteration.

- **Traffic Padding:**

  The addition of extra bits in to the gaps present between the data stream in order to discourage the attempts of traffic analysis.

- **Data Integrity:**

  All these mechanisms ensure the reliability of the data. i.e. during the transmission and when it is received at the destination end the data is same as it was when sent.

The other techniques which come under specific security mechanisms are Digital Signature, Notarization, Authentication exchange, Routing Control and Access Control.

Pervasive Security Mechanisms:

The mechanisms included here are those which are not fixed to any protocol layer or security service. Some of its types are as follows.

- **Security Audit Trail:**

  In security audit the data is independently examined and reviewed of the systems self examinational activities and records.

- **Event Detection:**

  It is to detect any security related issues, so that counter measures can be taken against that act or event.

- **Security Recovery:**

  It handles all the requests of restoring the system back from the damages caused by any illegal activity from all of the security mechanisms.

Rest of the securing techniques like Trusted Functionality and Security label are also included in pervasive security mechanisms.

There are many reasons to discuss this existing security frame work here. Off them the primary reason is this that our research is also about securing one type of network like MANET. Every network has its specific requirements of security but the entire measures move around the same basic frame work of 'SSS' i.e. Security attacks, then Security mechanisms and at the end Security service [4]. Similarly in this dissertation the measures are taken against application layer DoS attacks. First the attack then here the security mechanism i.e. containment of the attack is studied and then under the heading of security service many query distribution techniques are developed so that the effects of this attack can be minimized.

1.7 MANETs Security:

In this section we present some of the problems and issues which MANETs face. Also presenting the measures that have been taken to secure them from these issues. As mentioned in the 'SSS' frame work, similarly according to it in MANETs we have to make efforts to prevent, detect and recover the system from any malicious activity. In the first half of this portion we are familiarizing our reader with the security of MANETs then later on we have focused on the application layer DoS attack against the availability of Mobile Ad hoc Network. Hence minimizing its effects during the attack to let the network be available to its legitimate users.

Securing this infrastructure less network with the properties of Nodes failures, security vulnerabilities and unpredictable links is a great challenge. Hence they require efficient and effective security mechanisms and services to protect them against any attack. Some of the problems and their solutions are mentioned as follows [2].

MANETs face the problems of Redundancy and Robustness. To secure the network against them a Key management system is developed. This system works in corporation with intrusion detection mechanism, whose work is to report any malicious intentions. Thus the management system isolates those malicious nodes and the network is made secure.

MANETs are vulnerable to various kinds of active and passive attacks. Like ranging from disruption of services to eavesdropping. Conventional cryptographic techniques are not able to secure these networks against such attacks. Due to the dynamic, mobile nature of its nodes. So mobile agents are used with intrusion detection and prevention systems, which enhances these systems functionality and makes them to cope with the mobile nature of the nodes.

Fault tolerance is another problem faced by MANETs. That is to increase the immunity of the network against attacks like for example denial-of-service attacks.

We find that all the above mentioned solutions are either preventive, detective or they recover a network to restore its original functions. There is no such technique to contain the effects of an attack as the attack is prevailing in to the network. For this we need the security mechanism of containment. We need such techniques that prevent the whole network from over whelming. In our this research we have studied the Application Layer DoS attack and its effects on the network of denying the services to the legitimate users and producing starvation. We have developed the Query distribution techniques which work during the attack time thus not only providing fair distribution of resources among the users, but also preventing the whole network from getting down by minimizing the effects of the attack, and keeping the network functional while the attack has occurred.

1.8 Outline of the Thesis:

The chapter 1 of this dissertation is the Introduction, in it a detailed overview of MANETs is given along with it details about securing networks is presented. The chapter 2 is Literature Survey; it includes the previous work of researchers who have contributed to this domain. Chapter 3 is Problem Definition; it is about the aims of this research which are accomplished by conducting this research. The scope of this research is mentioned here. In this chapter the problem on which this research focuses is studied, all the requirements needed to cope with this problem are explained in details. Chapter 4 is proposed solution and methodology; in this chapter we have developed the query distribution techniques to provide fair share of resources and minimize the effects of the attack during the attack time and keep the network functional. In Chapter 5; we have presented the results which we have formulated after implementing the strategies in the simulator NS-2. Chapter 6 is Conclusion; in this we conclude our work together with giving some future issues which can be worked upon under the light of this research. At the end of this Dissertation we have added References; here we provide the list of links which have been consulted while carrying out this research.

2. Literature Survey

2. Literature Survey:

Research is a field with no bounds and limit, any one can contribute to this field who is a good thinker and is innovative. But with the emergence of new researches the work of pioneers who have contributed to this field can not be ignored or neglected. Similarly before conducting our this research we have studied books, thesis, different articles, research publications and other related material to determine what has been done in this domain earlier and to find out the gaps that are present in this specific area from which we have drawn our problem.

The literature survey is being divided into four main categories here which are as follows.

- **The first category includes:**
  Literature relevant to attack containment and application layer query flooding attacks
- **The second category includes:**
  The work which is being done on types of floods attacks in mobile ad hoc networks
- **The third category includes:**
  Work on few network layer flooding attacks in mobile ad hoc networks.
- **The fourth category includes:**
  Some of the work that has been done on containment in MANETs.

2.1 Attack Containment and Application Layer Query Flooding:

Daswani et.al. in year 2002 worked on application layer DOS attacks in Gnutella. They studied the effects of Denial of service attacks based on Query floods in Gnutella [5]. The authors divided the nodes into local nodes (clients) and super nodes (servers). Processing resources are pre-reserved for both types of nodes. The super nodes are core of the network and are responsible for routing and processing the incoming queries. When a client makes a query to the super node, it after conducting local search broadcasts the request to its neighboring super nodes. Similarly in the same fashion the malicious queries can grow exponentially in the network and consume all the resources and causing starvation at legitimate clients end.

They developed a simple Traffic model in the solution, and run simulations on it to see how different Load balancing techniques can be used to contain or minimize the damage caused by Query flood DoS attacks. To provide fairness authors proposed different incoming allocation strategies and drop strategies. In results authors calculate the damage caused by different malicious nodes under the combinations of proposed strategies.

Daswani et.al. in 2004 studied the application layer DoS attacks "the Blasting attacks" in Chord. Chord is a structured P2P distributed hash table network (DHTs) [6]. In these attacks the malicious nodes introduce heavy traffic in to the network, thus denying the network to service the legitimate queries. Thus the network becomes unavailable to the legitimate queries and unable to carry out its normal functions. This results in degrading network's performance.

In this paper the authors have proposed and evaluated different traffic management policies that could contain the excessive queries generated by malicious nodes in order to maximize the system throughput. Different Incoming Allocation Strategies and Drop down strategies are developed for chord and evaluated by running simulations to see which best works during these blasting attacks. Also these policies are able to remove the damages that have been done by these blasting attacks by eliminating the load of excessive queries from the network.

Daswani et.al. in 2004 made a model of GUESS (Gnutella UDP Extension for Scalable Searches) network which is an unstructured P2Pntwork and studied the problems of resource discovery in it [7]. In this network nodes contain pong caches which indicate those nodes that are available to accept the queries. Here the problem of pong cache poisoning in which there is an entry of malicious node in the cache has been highlighted and dealt so that its effects can be minimized.

Here a model has been presented by the authors which determine that how the pong caches are poisoned by the exchange of messages and how the caches can be recovered back, for this they have proposed solution to this problem which is ID Smearing Algorithm (IDSA) and Dynamic network partitioning (DNP) scheme. Authors also discuss the techniques to seed the cache of newly born nodes. The results show a good improvement under various types of attacks and malicious nodes.

Daswani et.al. in 2004 have described the widely usage of peer to peer networks today [8]. Peer-to-peer networks are comprised of an autonomous range of resources of computers (for e.g. like files, CPU cycles, storage space) which enable a P2P system to accomplish very costly tasks inexpensively. Also have described the problems associated with search and security of these data sharing Networks. Search refers to locate the required data, where as security refers to protective measures against different attacks.

The major focus of the authors is towards handling the flooding attacks. As Peer-to-Peer networks mostly use flooding algorithms to share information with each other. This provides an added opportunity to an attacker to launch an attack. Authors also introduce the idea of attack containment along with the attack prevention, detection and recovery. Yet the existing security measures are not enough to fully secure this open nature network. In this authors have presented those areas and measures which are to be worked in future and highlighted the problems needed to be addressed.

Mayank et.al. in 2004 have presented the overview of all the research done related to peer to peer networks at Stanford university [9]. In this report they have described all the issues and challenges that peer to peer networks face. Peer-to-peer networks have many security challenges to face off them one is Scalability. With increasing nodes in the network this activity brings lot more issues to be handled like the need of more resources, their distribution among the nodes, determining and identifying the locations of the peers and even the network is open to entry of any malicious node in the network. Hence the network should be made capable of coping with constantly changing requirements of resources and how these resources can be found and best utilized in peer to peer networks in order to provide maximum throughput.

The solution provided here is Resource Market Place; this encourages the incentives of sharing. This technique enables the network to determine the amount of resources that are needed in any specific situation. Thus with the changing network components the network is able to maintain and fulfill its requirements of resources.

Qixiang et.al. in year 2003 described in this paper that how the remote work in flooding based peer to peer system like Gnutella can be maximized with controlling the Query injection rate at each node in the network [10,11]. Because due to flooding nature of nodes any useless or malicious query can exponentially grow in the network resulting in occupying all the resources, hence due to which the legitimate queries do not get their share of resources. As a result the network is not able to produce its desired output.

Here the term remote works mean serving the other nodes of the network. If a network is serving more and more nodes and information flow is smooth and there is no starvation or conflict among the nodes of the network than the network is said to be efficient. The authors focus to allocate resources among all the nodes fairly so that the network achieves high quality of service. This is achieved by controlling the query injection rate at each node.

2.2 Flood Types in MANETs:

Brad Williams et.al. In year 2002 have presented a detailed comparison based study of all those broadcasting techniques that are in use of Mobile Ad hoc Networks [16]. Due to the wide use of MANETs now we need standard methods and route establishing techniques to efficiently deliver a packet from one node to another. So this research of comparative study among routing techniques helps a MANET to choose a suitable broadcasting technique according to its own network conditions.

All the protocols are being classified and categorized in to four mentioned techniques that are: Simple Flooding, Probability Based Methods, Area Based and Neighbor Knowledge Methods. From each category one or two protocols have been tested and evaluated for their performance through simulations under various types of networks and there respective conditions. The graphs portray the results of above conducted analysis that is which protocol is best suited under what conditions with which broadcasting technique.

Rudiger et.al. in year 2002 have presented the detailed study comprising of similarities and differences in two types of networks i.e. Mobile Ad hoc Networks and Peer-to-Peer Networks. So these networks could be clearly understandable [25]. This paper gives a clear discrimination among these two networks, so the selection of which type of network under specific terms and conditions are easy to make.

Both of these networks are decentralize and self organizing. They have similar concerns regarding network management policies and routing strategies. This is mainly because both of these networks have to serve a decentralized and completely unmanaged environment. Both use flooding based protocols for packet dissemination and many more similarities are discussed here. The primary difference in these two networks is that they based on different networks. The Mobile Ad hoc networks are based on radio network where as Peer-to-Peer on IP network. Nodes of both networks have different nature (for e.g. nodes in MANETs are mobile while in P2P they are fixed) Bandwidth issues are different. Both have different protocols of routing. Many more differences are discussed here. While under the light of this comparison the main focus of this paper is this that how queries are routed in these two networks and how different routing algorithms are employed in both.

2.3 Network layer Flooding Attacks:

Chi- kin chau et.al. in year 2008 have presented the idea of inter-domain routing for Mobile Ad hoc networks [22]. Inter -domain routing is the main element that plays a very important role in communicating different networks that are under different organizations. Inter –domain routing has flourished very well in internet but it had to face many challenges incase of Mobile Ad hoc networks.

In MANETs this problem is challenged by the mobile nature of nodes, which raise the dynamic network topology issue. The next challenge is the variety of intra-domain routing protocols of MANETs. In this paper the authors have not only coped with these challenges but also have developed an efficient inter-domain routing framework for Ad hoc networks.

Sasu Tarkouma et.al in year 2008 have worked on the issues related to end to end communications with in a network [23]. Here we have a data and interest based traffic forwarding network architecture. In this paper the network appears as a Black box to those utilizing it.

Black boxes hide the network architecture from the outsiders and exposes only that information to outsiders which is off there interest and use, and hide the rest network details from them. Hence this proposed architecture enables to have more secure and reliable end to end communication even in distributed networks as not allowing the outsiders to see the details of the network not related to them.

Bjorn Scheuermann et.al in year 2008 have worked on the problem of congestion control in multicast traffic in Mobile Ad hoc networks [24]. Multicasting helps in delivering the packets to multiple hops thus enabling to save the limited bandwidth of Mobile Ad hoc networks. But it faces the issue of congestion.

In this paper the authors have proposed a hop-by-hop congestion control concept. This concept has been implemented by using the Backpressure Multicast Congestion Control (BMCC) protocol together with geographic multicast routing protocol the Scalable Positioned-Based Multicast (SPBM). By using NS -2 simulations the proposed concept is being evaluated and tested for performance.

2.4 Containment in MANETs:

Robert Castaneda et.al in the year in 2004 has worked on one class of routing protocols known as On-Demand Routing protocols to reduce the routing overload from the network [26]. These protocols determine a route by using query flood. A query in search of the route is flooded over entire network to determine source to destination path. Each query has unique identifier, which prevents same query from same node to be multiply propagated. Hence when destination receives this first query it responses back with a route reply packet. Then later on all the traffic to the destination is made over this determined route. But this yet leaves routing overload over the network as it floods route query in the start while determining.

The solution to this problem to reduce more routing load from the network is presented in this paper by enhancing this technique and introducing the concept of Query Localization i.e. to the limited region of the network this paper has Localized the Query flood. This approach makes use of route histories to contain the query flood to a limited region and does not require any location information, as it restricts the query flood only to its neighboring nodes.

This approach works efficiently and increases network performance by removing more routing overload then earlier. The results via simulations are made by testing and evaluating this approach on the on demand routing protocols to see the remarkable performance of Query Localization approach.

The author Abdelhafez Mohamed Abdelfattah in the year 2007 in this dissertation has worked on mitigating techniques [27]. He has studied the impact of worms and there effects. As we know worms replicate and expand frequently and tremendously once they have become active. To have preventive and defensive mechanisms against them is a great security challenge. There are many existing techniques developed to control worms but there effects are not being tested and analyzed on any running traffic.

This thesis has made three contributions to this domain. The first contribution is this that it has provided a comparative analysis of performance and the effectiveness of all the existing defensive measures by running them on live traffic. It has also introduced another technique which is more efficient and it also mitigates the spreading rate of worms with TCP connections working for infiltration. The second is presenting a detailed comparative study among the worm Flash and its variant Compact Flash (CFlash) through simulations showing that with same parameters the new worm's propagation rate is more increasing. The third contribution is this that the behavioral study of TCP worms in MANETs. He has developed an Analytical Model which shows the parameters required by these worms to spread then solutions for the model to minimize there effects, verifying the results through simulations.

Bo-Chao Cheng et al. in the year 2008 have introduced the concept of containing of malicious effects in the existing intrusion detection systems (IDS) [28]. Intrusion Detection systems are developed to detect any malicious nodes activities in MANETs. But due to in appropriate mechanisms of IDS response, the best IDS even can not get the desired results, because as they are not rapid in conveying the malicious nodes entry in to the network.

This paper enhances the functionality of IDS by introducing the concept of containment strategies to limit the degree of an attack. Paper has presented a T-SecAODV protocol which gives rapid responses, thus it disseminates the malicious nodes information rapidly to all the nodes. These alerts enable the nodes to reset their routing tables so that they can easily detect and isolate the malicious node from the network thus the effects of the attack are minimized. Simulations are run to see the performance of this protocol.

2.5 Limitations in Literature Surveyed:

Some of the discrepancies that are seen in this survey, which would be addressed in our research, are as follows:

There are different types of network that are under study and lots of research is being carried out in the area of security of these different networks. The above dissertation is only about fixed peer to peer networks [5-11]. The Mobile Ad hoc networks are not being addressed here. It has been identified here that how application layer denial of service attack can be damaging to peer to peer networks. Techniques for containing these attacks in P2P networks are introduced in this research. But effects or containment of these attacks in Mobile Ad hoc networks are not being mentioned here.

Every peer to peer network is constructed on the bases of some fixed topologies. We see a fixed topology in its structure. In the above research [5-7] all the attack containment strategies or techniques that are being formulated in the proposed solution are fixed topology based. Incase of Mobile Ad hoc networks we see no such fixed topologies. Rather they are dynamic topologies based networks. So these strategies become in applicable to these networks.

In Ad hoc networks as all the nodes are mobile, so they require battery power and energy to remain active. While in peer to peer networks we see no such requirement needed. So all the proposed strategies are formulated without considering the issue of battery power and energy consumption [5-7]. This cannot be ignored in Mobile Ad hoc networks. Hence for the containing strategies to be successful, these should be cost effective not only in terms of performance but also in terms of less battery power and energy consumption.

In this research the traffic management techniques or strategies are based on one assumption made that the legitimate nodes have the capacity to see all the queries arriving on their particular links [7]. This assumption cannot be made in the case of Mobile Ad hoc networks because due to nodes mobility, secondly as we know that these spontaneous networks nodes spent battery power and energy just to see the arriving queries. If the rate of incoming queries is high at a link then the node might drain out its battery just in order to for see all the incoming queries. So there is a need to reconstruct these traffic management policies and models in order to contain attacks in Mobile Ad hoc networks.

In this paper author has defined the remote work as "serving other nodes queries" [10, 11]. When the other nodes query is being propagated, we do are serving the node but no remote work is done. It is only done in the case when the other node's query is not only propagated but is also being answered. Hence in this paper author has not made any differentiation among answerable and forward able queries.

2.6 Summary of the Chapter:

MANETs are emerging and popular network of today. Their importance and applications are increasing rapidly. Due to their wide usage especially in the situations when networks are to be created on the fly under critical circumstances this is the most efficient system to be build. But this enormous growth of these networks also requires strong, efficient and effective security measures to protect them against malicious activities. The earlier security measures presented in the literature are not enough to cope with Application layer Query Flood DoS attacks in MANETs. It requires such efficient security measures which not only can Prevent, Detect or Recover the network from the damages caused by these attacks but also can contain these damages to let the network remain functional even in the light of these attacks when they are in action.

3. Requirement Analysis

3. Requirements Analysis:

MANETs are a subset of wireless networks; they are today's widely acceptable and progressive networks. They are gaining popularity since last decade. They are becoming mature day by day but there practical implementation is quite difficult due to challenges in security. The open, dynamic, infrastructure less and energy constrained nature imposes sensitive, efficient, and robust security measures to be developed for these types of networks. In order to develop security measures for this type of network we have to make a study of the traditional security objectives which are to be satisfied with the particular needs of the MANETs.

The security goals or objectives which we are predicting for MANETs can be thought as an extension to the objectives which are set for traditional networks. According to these the security should be provided in terms of Confidentiality, Integrity, Availability. Authentication, Non-repudiation, authorization and Access control as already discussed in chapter 1, [29]. But apart from these the other factors which must be satisfied due to the inconsistent and un-predictable nature of MANETs are physical security, fault tolerance, privacy, limited computational complexity, Energy and power consumption and lot more [29]. There are two more essential issues which can not be compromised with security. The security measures are always influenced by these two attributes which are Cost and Usability or Utility.

We are categorizing our analysis or requirement in to three main aspects of study which should always be conducted while developing security mechanisms not only for +MANETs but for other networks as well. Firstly a little brief survey of the traditional Attack classification Then we are giving a brief study of attacks on the different Layers i.e. which types of attacks occur at each layer and then finally those specific features of MANETs which massively effect security and make it challenging task to be accomplished.

3.1 Categorize of Analysis:

- ✓ Classification of Attacks.
- ✓ Types of attacks at each layer.
- ✓ Challenges imposing features of MANETs in way of security.

3.1.1 Classification of Attacks:

The use of Mobile Radio technologies are rapidly growing for Data and Voice communications, generating many new networking concepts among them the most acceptable and penetrating short term networks is the MANETs. In order to create a robust, efficient and fault-tolerant security structure we need to deeply understand the threats which are possible along with their consequences. Here we have discussed all the wired network's attacks to which MANETs are also vulnerable. The attacks are classified in the following table [29], [30], [31].

	Threat	**Generated by Attack**
Active	Denial of Service (DoS)	Worm hole Black hole Selfish Node Sink hole Route Error Falsification Query Flood Gray Hole
	Masquerade	Spoofing Sybill
	Modification	Altering Routing Tables Loop Forming
	Eaves Drop	Traffic Snooping
	Jamming	Traffic Subversion

	Fabrication	False routing Messages Fake Packets
	Replay	Rushing Short Circuit
Passive	Release of Massage Contents	Hacking or Snooping
	Traffic Analysis	Traffic analysis Silent Node Exposure

Table 3.1: Classification of Attacks [29], [30] and [31]

3.1.2 Types of attacks at each layer:

In MANETs Per layer threats are classified as follows i.e. against physical, Medium Access Control, Network and Application Layer [32].

- ✓ At physical Layer:

  The link is only jammed by the jammer by forcing un-necessary bits to the link, but it does not misuse any feature of communication protocol.
- ✓ At MAC Layer:

  Jammer not only jams the link but it also misuses the MAC protocol's features.
- ✓ At Network Layer:

  The adversaries try to disturb the link establishment, neighbor discovering (i.e. incorrect forwarding tables) functions of the network and other network securing services.
- ✓ At Application Layer:

  The attacker attacks the distributed communication capabilities like attacks on Data Aggregation, in it's or Source Authentication and other properties.

Attack at any one layer eventually disturbs the functionalities of other layers due to the Cross- layer network designs. If the attacker attacks on the neighbor discovery attribute of the communication protocol at the Network layer hence it would miss-route the Application layer data too.

3.1.3 Challenges imposing features of MANETs in way of security:

Some of the significant features of MANETs which pose challenges as well as provide opportunities to achieve security objectives are as follows [34].

- Due to the de-centralize nature of MANETs, they also require distributed architecture security measures so to attain longer survival.
- The security measures should adapt on-fly the changes like frequent joining and leaving of nodes, nodes having no prior relationships among each other etc. These mechanisms should be able to cope with the dynamic nature of nodes.
- A MANET may have ten; hundred or even thousands of nodes in single duration of time, so scalable security measures are needed to be able to cope with such large network.
- Security measures also have to cope with the MANETs issues of scarce energy, limited computational complexities and resources also the in-persistent network structure.
- Wireless links make MANETs vulnerable to links attacks.

One of the major threats to MANETs is the Denial-of-service attack. Lots of research has been done for prevention against this attack at MAC and Network layers [33]. Some of the work regarding this attack on Application layer has been also done. But off the attacks done by this threat the Application layer Query flood DoS attack is the most damaging one. Because a single Query transmitted by a node can be received at more than hundreds of nodes at third or fourth hop. If this query is malicious then it would be flooded over the entire network which would then result in wastage of processing resources and starvation at the legitimate user's end, and if this malicious activity is left un-noticed then this effect very soon will prevail in the network and overwhelm the entire network.

Hence due to lack of any fair distribution of resources mechanism a single node can occupy the entire network resources, the rest of the nodes will not be able to get their share of resources. As there is no mechanism to mitigate the effects of attacks during the attack time, which result in damaging the whole network and disallowing the network to serve its legitimate users and whole network gets down and its performance is degraded as it becomes unable to output its desired results.

3.2 Problem Definition:

We have categorized our Problem domain in to two aspects one is Flooding and the second one is Single/Multiple Attackers. The attacker can use any of the flooding types. All the problematic situations associated with each aspect are discussed in detail.

3.2.1 Flooding:

To share information with each other the nodes of the ad hoc network use flooding algorithms. There are two main types of flooding Simple/Blind Flooding and Intelligent Flooding. Here we have focused and studied all the problem scenarios which are associated with flooding and its types.

Flooding algorithms send the single query to all the directly connected nodes. If this flooding continues up to three or four hopes than a single query can be received by more than 100 nodes. This geometric increase provides added opportunity to attackers for launching a query flooding attack by generating only few queries. Figure 3.1, 3.2 and 3.3 explains the scenario in detail.

The one common solution that comes into mind immediately after listening this problem is not to use flooding algorithms. But the research indicates that the use of flooding algorithms is necessary for ad hoc network because of the reasons specified in the section 1.3 and 1.4 of Chapter 1. And our research also justifies its use in MANETs.

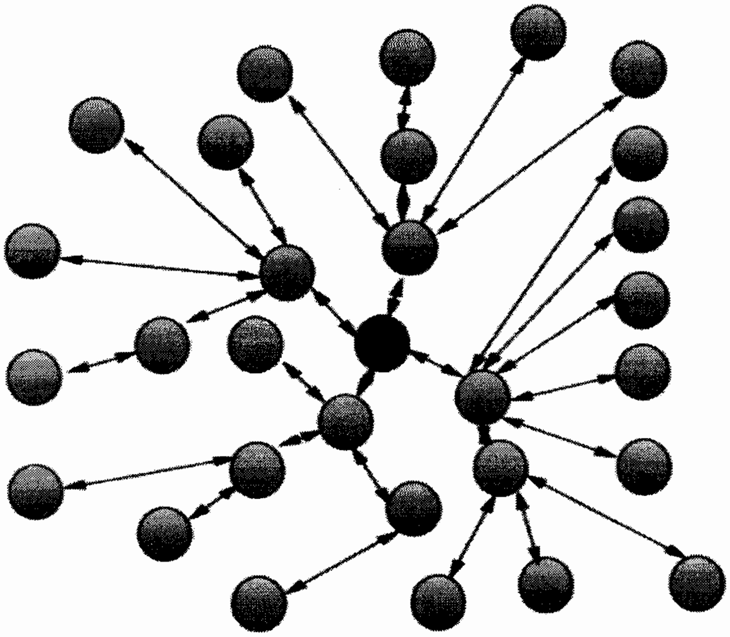

Figure 3.1: Node N wants to send a Query

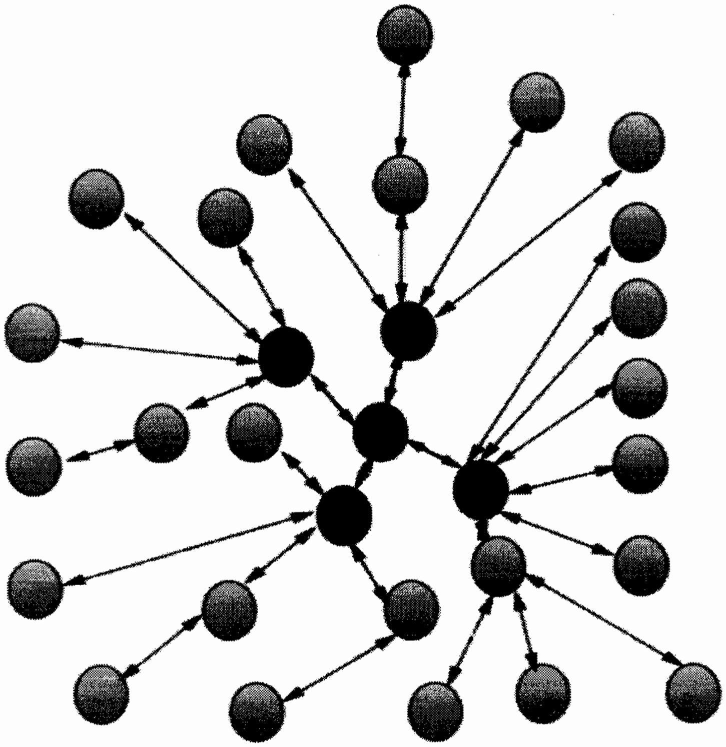

Figure 3.2: Node N's Query at First Hop

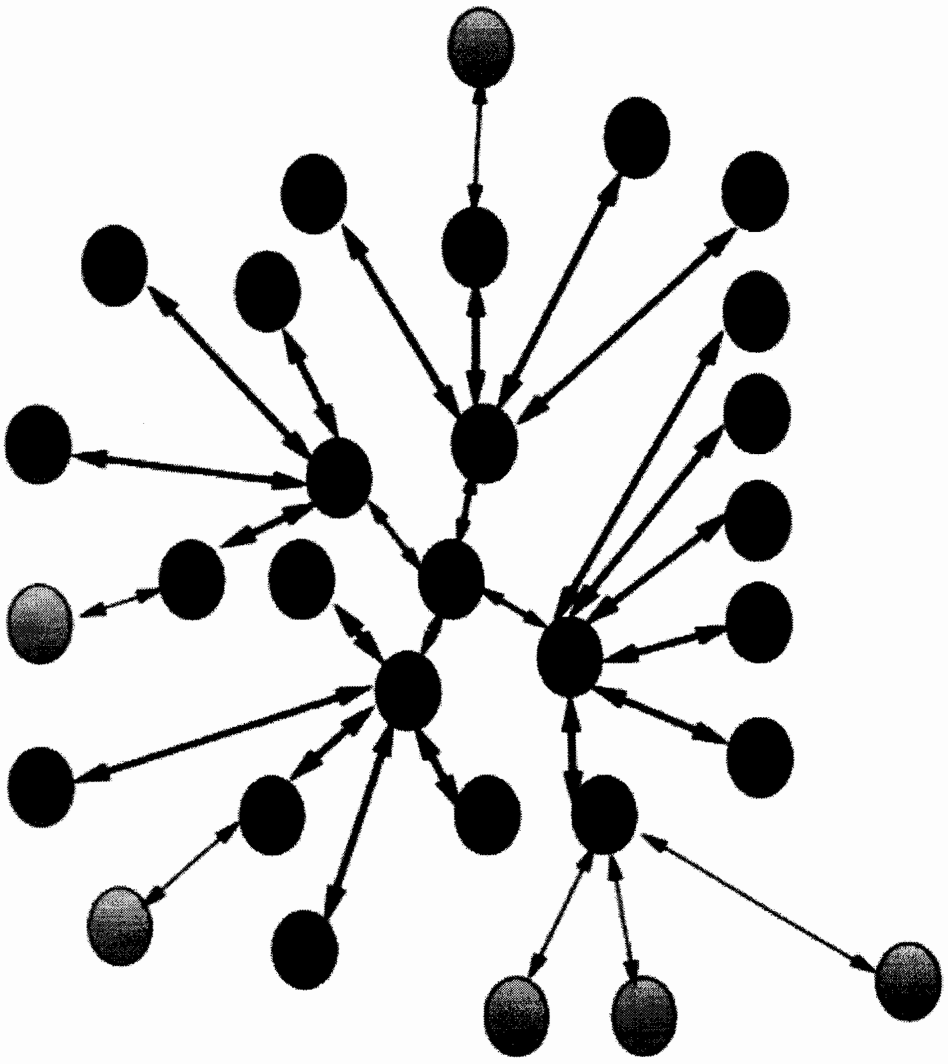

Figure 3.3: Node N's Query at Second Hop

3.2.2 Types of Flooding:

- **Simple/Blind Flooding:**

In this type of flooding an attacker simply uses all of its processing resources to send as many queries as possible without taking notice of how many queries a victim can process. This can be easily controlled by eliminating the node which is continuously generating excessive queries to the network hence the attacker is blocked.

- **Intelligent Flooding:**

In intelligent flooding the attacker tailors its attack against the particular query distribution techniques that a legitimate node is using. This is more hazardous because instead of generating a blind flood an intelligent attacker can select different kinds of flooding strategies to achieve maximum output. If we analyze it from victim point of view then different flooding strategies can results in different level of damage. Now the question is that which flooding strategy an attacker can use in different scenarios. Therefore the formulation and evaluation of these flooding strategies and their counter measures is also an open issue that need to be resolved.

3.2.3 Single/Multiple Attackers:

Every node has a processing limit and can serve limited number of queries per unit time. The attacker is needed to send only as many queries as the victim can serve within a particular period of time. This situation will results in serving the queries of an attacker only and will waste the network resources as well as results in starvation at legitimate user's end. The victim will only serve the queries of the attacker. The queries of the other users will not be entertained.

Besides good attack prevention and detection techniques there is no solution to minimize the effect of an attack during the attack time. In short ad hoc network does not have any attack containment technique. Minimizing the loss and wastage of resources during the attack time is still an open issue in ad hoc networks. In the absence of any attack containment technique a single attacker can disturb the whole network and can consume maximum resources of the network. The scenario is explained with the help of the figure 3.4.

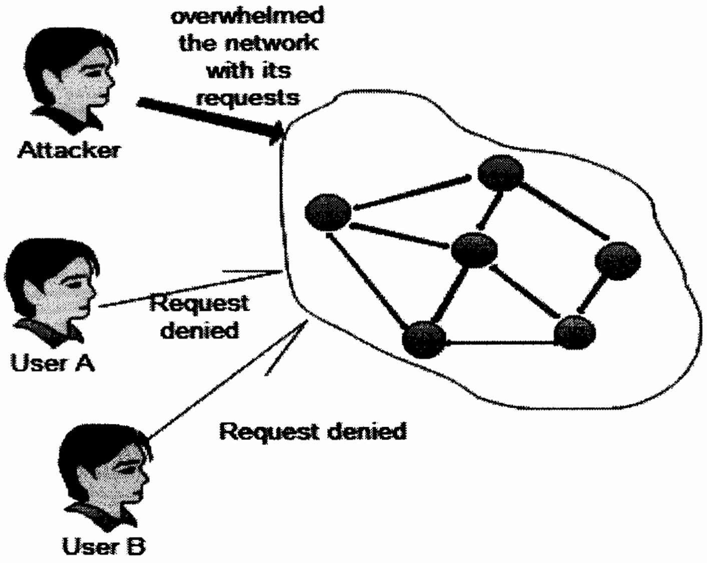

Figure 3.4: In the Absence of Attack Containment Technique Single Attacker can Overwhelm the entire Network

A network at a time can be attacked by more then one attacker at a time. Multiple attackers can take different positions in the network. They can work either individually or mutually and send as many queries according to the processing limit of the nodes. As we know that a single attacker itself is damaging, so multiple attackers with different flooding strategies can bring greater loss to the network. If for example a network is comprised of ten nodes and each node is attacked by an attacker then within a single second the whole network will get down and all of the legitimate users will starve, a complete wastage of all the network resources and un- availability of network. Thus a mechanism to control the query injection rate at each node is still an open issue. The scenario is explained with the help of figure 3.5.

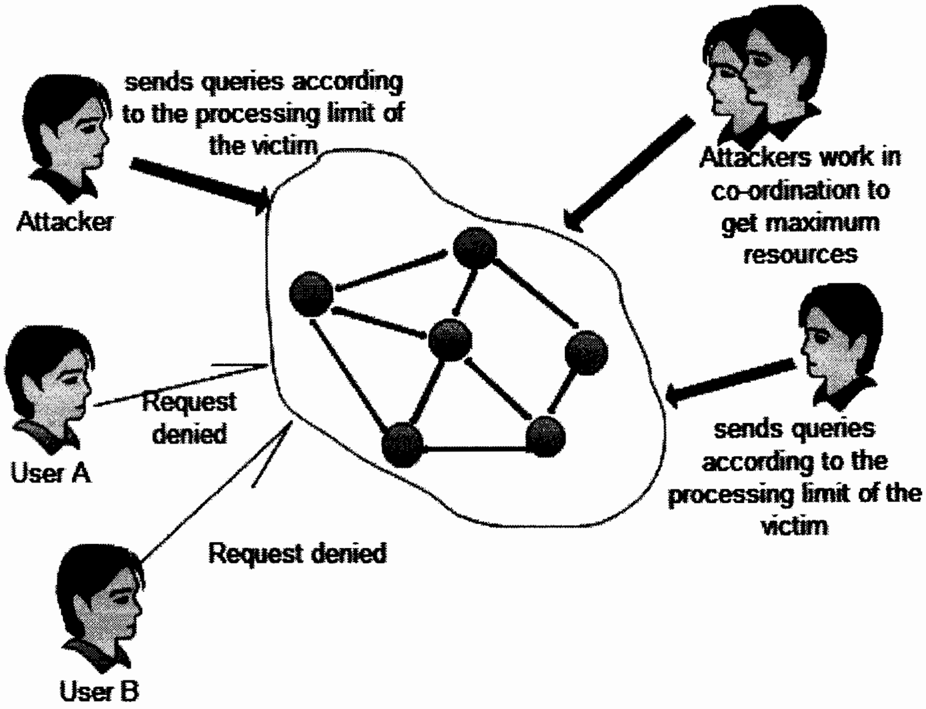
sends queries according to the processing limit of the victim
Attacker
Attackers work in co-ordination to get maximum resources
Request denied
User A
sends queries according to the processing limit of the victim
Request denied
User B

Figure 3.5: With no mechanism to Control Query Injection rate at each node, Multiple Attackers occupy the entire processing resources at each node.

3.3 Objectives of the Research:

According to the different problematic situations discussed in section 3.2, we have set certain goals of this research which have built in solutions for the problems already discussed. The targets and objectives of this research with underlined methodologies are as follows.

3.3.1 Fair Distribution of Resources:

To achieve the fair distribution we proposed different query distribution strategies which allocate the resources on the basis of specific mathematical models. Hence the monopoly of any single node over the entire network is discarded. With the help of these techniques all the nodes mutually share all the resources. Nodes can not behave selfishly (i.e. nodes denying participating or sharing).

3.3.2 Enhancing Security:

We have proposed these techniques which not only fairly distribute the resources among the nodes but also contain the effects of the attack during the time the attack time, as currently MANETs lack such attack containing mechanism. This prevents entire network overwhelming.

3.3.3 Minimizing chances of Starvation:

As the resources are fairly distributed among the nodes i.e. requests from all the nodes are taken and processed, hence any of the malicious nodes who have entered become unable to occupy all the network resources, thus legitimate users get their network share for processing their requests. Hence are prevented from starving.

3.3.4 Keeping network functional even in light of an attack:

These containing strategies or techniques reduce the effects of the attack during its attack time, thus allowing the network to remain functional under the effects of the attack, because as the chances of starvation is minimized. The network is allowed to produce its desired results.

3.3.5 Justifying the use of flooding algorithms:

As the flooding nature of protocols provide an added opportunity for an attacker to launch an attack, as a single malicious query is flooded over the entire network. One straight solution is to stop using these algorithms. But our research with the help of these techniques will justify their use in MANETs.

3.3.6 Consuming less energy and processing capabilities:

In MANETs energy and computational complexities are constrained. So while considering these two limitations the distribution techniques have been proposed here are based on simple mathematical models, which require simple processing capabilities and also consume less energy and time to operate.

3.3.7 Cost effective and enhanced usability:

The proposed solution is more cost effective as techniques developed consume less energy. The proposed security measures enhances the networks utility as it prevents the entire network overwhelming and even allows the network to do legitimate work even in the light of an attack.

3.3.8 Evaluation and Formulation:

Each of the strategy or technique is formulated and evaluated against the different flooding strategies by using NS2.

3.4 Summary of the Chapter:

There are many different ways through which the surface of the MANETs can be attacked. Off them the most critical is the Application layer query flood DoS attack. In which an adversary or a group of adversaries can contaminate the network by the injection of just single or few queries and deny the services to the legitimate users. This main problem has been studied with all of the damages it can cause to a network. It is based on two main aspects one is the Flooding techniques and the other is Single/Multiple attackers.

The blend of these two aspects brings damages like Starvation, wastage of resources, wastage of individual nodes processing capabilities and other damages are encountered. With these damages the emerging issues like need of attack containing, fair distribution and similar other needs are projected by this study.

Security usually imposes restrictions on the availability of the network or the system but we here infect by controlling the query injection rate at each node thus to deny the intentions of a single node or multiple nodes to capture all the resources and through provision of fair distribution, Thus make the network to be available to its legitimate users even in the light of an attack.

4. Proposed Solution

4. Proposed Solution:

In the previous chapter we have deeply explained our problem which is Application layer query flood DoS attack together with all of the associated damages this attack can bring to Mobile Ad hoc Networks. MANETs are more vulnerable to this attack because of the inherited built in disadvantages from which they suffer like bandwidth and energy or power constrained, poor and no centralized network management due to dynamic topologies and limited security [29]. These are also some security implementation constraints to this wireless radio technology based network. Our provided solution caters with the problem while considering these issues also.

By considering the above mentioned problem we have designed our proposed solution to counter back this attack and also reduce the damages caused to keep the network functional during the attack time. A MANET is composed of set of mobile devices those coming under the same radio range, they communicate with each other directly without the need of any access points. Whenever a user submits a query to a node, if that node cannot fulfill the request, the node passes the request to the next wireless devices or nodes within the same perimeter which act as intermediate switches; hence the query is broadcasted over the entire network. Thus such large number of queries either from the legitimate nodes or from the malicious nodes will grow exponentially with in the network and occupying all the resources and denying them to rest of the users. This would result in starvation at many users end.

In our thesis we formulate different fair distribution strategies which attempt to provide a "fair share" of resources to all the nodes, hence making it difficult for the malicious node or the attacker to deny services to legitimate users or nodes and thus prevent starvation. Till to date all the work that has been done on DoS attack is not regarding this discipline which we have focused on. All the exciting work comprised of either recovery oriented techniques, in which the on going attack is detected and resources are denied to the adversaries or preventive techniques in which resources are prevented to be accessed. In contrast to these techniques our proposed solution does not require the nodes to distinguish malicious queries from the good (bona fide) ones, indeed the malicious nodes or attacker will get some of the share of the resources. But our proposed fair distribution techniques or strategies make sure that the malicious nodes do not get the excessive share of the network and its resources.

Our fair distribution strategies with the aim of balancing the load at each node and hence over the network together with containing malicious effects, do not remove the need of recovery or preventive techniques. Although we believe in order to provide protection against DoS attacks all these types of security measures are needed. But as work in respect to fair distribution of resources and attack containment has been not studied so it is the focus of our research. Thus in a Mobile Ad hoc Network it is a great challenge to maintain fair distribution due to the exponential growth of query flood.

In the beginning of this chapter we have given a brief introduction of our proposed solution. Now here we are presenting the detailed working and functionality of the solution.

To understand the solution clearly we here take an example of simple network comprising of just four nodes with an assumption that each node has a single user of its own. While in real scenario a single node can have single or multiple users at a time. Each node can process say 'Q' queries at a time. Users can put up queries at their own node for processing; in return these queries can be forwarded to other nodes in the network to be accomplished. For instance if user at node B issues a query and its own node is not able to solve it then node B would forward the query to the rest of the nodes in the network. Hence all the nodes would be indulged in working on this query and will reply back also to node B. The figure 4.1 shows the scenario.

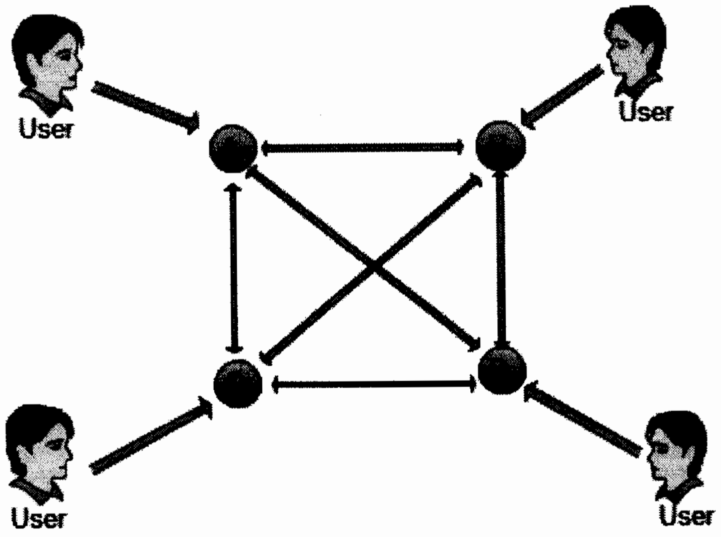

Figure 4.1: An example of four Node Network

We assume here that for instant each user has ¼ Q queries in per unit of time. When ever a user issues a query to its node at the same time that query is forwarded to all other three nodes of the network. I.e. each node will receive ¼ Q queries from its on user and ¾ queries from rest of the three nodes in the network. We can say that this is the situation where each node is executing a maximum amount of queries it can process in per unit of time.

Here we consider a scenario in which any one of the nodes say node A is malicious. Being malicious this node can attempt a variety of attacks to the network, but as our focus is denial-of-service attack, which can be conducted easily by any node. Node A can easily conduct the DoS attack simply by issuing more queries than ¼ Q queries, and flooding them towards the other nodes with the target of wasting the processing resources at the nodes B, C, and D, hence denying the resources to legitimate queries. In the above case when node A issues more queries then the desired set capacity. Then due to this increased load caused by A, node B, C and D will receive excessive queries from A which they can handle, hence they would have to drop some of the legitimate queries to cope with this excessive load of queries generated by the malicious node A.

In real network the above scenario can become more complicated as the nodes B, C and D would not be able to distinguish between the traffic i.e. either this excessive traffic is generated by this malicious node A or it has started forwarding the traffic on behalf of other network nodes other than those shown in the figure. As MANETs use flooding based protocols, so nodes in this network broadcast the queries to their neighboring nodes, hence a single malicious query could exponentially grow in the network capturing all the query processing resources, while denying these resources to many of the legitimate queries of the authorized users thus with the wastage of resources also causing starvation at legitimate end and can also even paralyze the whole network if this malicious action is left un-controlled.

One simple and straight forward solution that comes to mind is to stop using flooding based protocols but in case of MANETs flooding is the only option for attaining optimal output in this type of wireless Ad hoc networks because of the reasons as specified in the section 1.4 of chapter 1. In which the importance of these protocols is clearly explained.

Now in our above mentioned scenario the nodes B, C and D will have to use some kind of "queries distribution policy or strategy" to cope with this excessive load of queries through which these nodes will have to accept some of the queries and drop the rest according to their processing limits through fair distribution of resources. So every node in the network requires this strategy or technique to control the rate of query injection at each node to control the burst of queries. Since as the nodes B, C and D would not know to which other nodes they are connected to in the network i.e. either they are malicious or they are good nodes.

So here for instance they would simply decide to fractionally divide (i.e. fair distribution) their processing capacity i.e. Q/4 to each of the node from which they are receiving the queries and reserve the same ratio of Q/4 for their own user's queries. Doing so would reduce the legitimate queries but would also minimize starvation as giving chance to every node to have its share of sending queries. It would also reduce the number of malicious queries to be processed and thus contain the effects of the attack during the attack time and prevent whole network from getting down. These fair distribution techniques will also work as attack containment techniques here.

We have just explained the concept of our solution by the simple study of fractional division. Later in this chapter we would study many other different query distribution strategies which nodes might use to choose the number of queries to be processed. Some times the query distribution strategies which nodes may choose by making bad decision would show the maximum success of malicious nodes while some chosen by making good decision would show that of legitimate nodes, so it would be necessary to choose such strategy that attempts to maximize the legitimate work by minimizing the processing of malicious queries.

An important task is here that through these query distribution strategies we would be introducing fairness along with it we would attempt to balance the load originating with in the network mainly due to the burst of malicious queries by adversaries and thus containing the effects of the attack too. An opponent can use different flooding strategies either blind or intelligent to tailor its attack for attaining maximum share of the query processing resources. More precisely we do not want the legitimate nodes to distinguish malicious queries from the good ones. We just want to provide fairness among each of the node in the network with the desire of sending a query.

The query distribution process works at a node as follows. Firstly the node assign's a "quota" to each of its link from which it will be receiving the queries. The quota to each link will be provided from the preset capacity "Q" (the total queries to be processed by a node) of that particular node. Secondly if the links generate more queries then the assigned quota then the node selects queries on First-come-First-served bases process to meet the links assigned quota. We call this phenomenon as a "query distribution or allocation strategy" or even "attack containment strategy". We would be studying these different strategies with both single and multiple attackers attempts.

With a more general example we can understand the working of this process which is as follows. In order to restrict the attacker during the attack time it is necessary to provide him only a small portion of network resources and victim's processing capabilities. If the resources are distributed fairly among the nodes of the ad hoc network than the attacker can not disturb the whole network and is not able to occupy a major share of the network resources.

To achieve this target we in this project will formulate different kinds of strategies to fairly distribute the resources among the nodes of the ad hoc network. If a node N can serve 100 queries per unit time (Q) and there are 10 nodes that are sending queries to node N then instead of serving blindly. The node N will allocate a fair share to each of the 10 sending nodes. Now if one of the nodes from these 10 nodes is an attacker then it will get only a small portion of the processing power of the node N in return a single attacker cannot overwhelm the whole network. Thus this would mitigate the malicious effects and minimize chances of starvation. Now here the question is that how to distribute the resources fairly among the senders of the queries? To answer this question we proposed different Query Distribution Strategies. Figure shows the detail architecture that how query distribution strategies work.

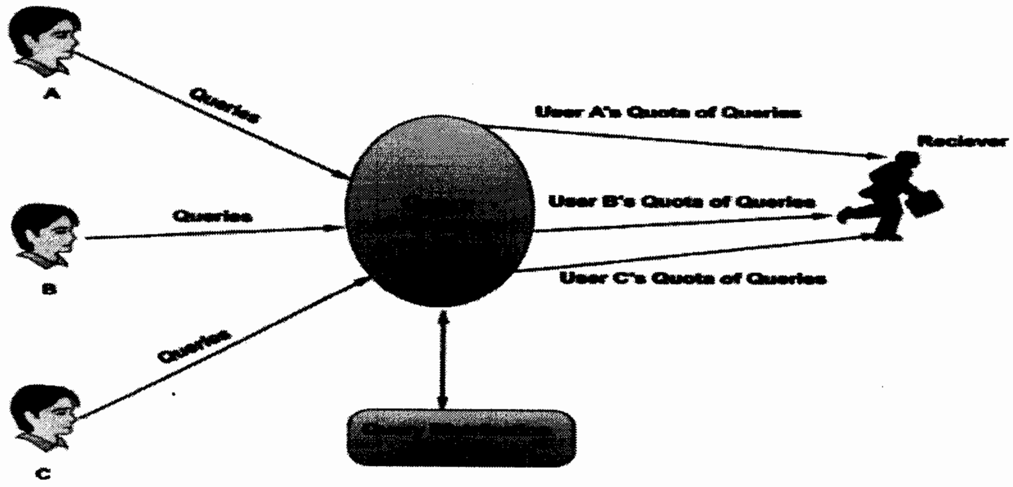
A
Queries
B
Queries
C
Queries
User A's Quota of Queries
User B's Quota of Queries
User C's Quota of Queries
Reciever
Query Distribution

Figure 4.2: Proposed Architecture

4.1 Query Distribution or Allocation Strategies (QDS):

In order to reduce the effects of Query-flood DoS attack the most crucial step a node has to take is to decide which queries to process and which queries to drop. If in a case a node processes too many of malicious queries generated by some malicious node. Then that node would be badly affected by this attack because this node than want be able to process the queries issued by the legitimate nodes. Such node would not only be wasting its own processing resources, but this node might forward these queries to other nodes thus now rest of the nodes would also have to decide to whether to process these queries. This would result in wasting more network processing resources. Due to the replicative nature of flooding a slight increase in the quantity of malicious queries can cause a massive waste of processing resources among the whole network.

So in this section we formulate various query distribution strategies with an aim to mitigate the replicative effects of query floods. Query distribution strategies (QDS) manage the load of burst of queries in the network after these have been issued by the malicious nodes. We are not making any discrimination among the two types of queries i.e. good queries and the malicious queries. The main objective which is to be achieved with these strategies is this that a slight increase in the quantity of malicious queries would result in not more than a slight drop in the performance of the network.

We have studied these different query strategies deeply here in this section in respect with two aspects of flooding i.e. firstly with Blind flooding secondly with Intelligent flooding. Also evaluating each of the strategy's performance in light an attack either conducted by a single attacker or multiple attackers.

4.2 Query Distribution Strategies against Both Types of Flooding:

In this section both the types of attacker use blind flooding to conduct an attack i.e. they simple send the burst of queries to capture maximum network resources to maximize their effects. In this case the attacker is simply busy in sending more and more queries at a node without knowing the query distribution strategy which that node might be using. We have developed different scenarios to see the working of these strategies.

In intelligent flooding both the single and the multiple attackers tailor their attack against a particular strategy that has been implemented at a node for fair distribution of query proceeding resources. That is the malicious nodes before conducting the attack know the applied strategy and then work according to it so to maximize there effects and attain maximum resources of the networks to remain successful in there intentions.

All the developed different scenarios below would also be tested against intelligent flooding to see the working of these strategies against intelligent attacks. In the next chapter we would present all these details, here for the understanding the working of these different query distribution strategies is analyzed with simple flooding in detail.

Strategy I:

Weighted Query Distribution Strategy with Single attacker:

This strategy works in a simple fashion i.e. the more you send the more quota you get for processing the queries. The link quota to the sender is assigned proportional to the number of queries that are arriving from that link. The number of queries if exceed to the allocated proportion of the link then the queries are chosen on first come first served bases.

Formulating it generally to have better understanding i.e. If a node N has ‘L’ links sending queries to it and it receives 1, 2, 3……….n number of queries from a single link. Then Share of Each Node = $\frac{n}{Total\,no\,of\,queries\,from\,all\,L\,to\,node\,N} \times Q$ hence the amount of queries from each of the link will be selected according to this formula.

Mathematical Calculation of Each Node Share through Weighted QDS:

Each node knows its processing capacity that how much queries it can serve during a particular period of time which is denoted by symbol Q.

After a particular time each node counts the queries residing in its incoming buffer.

It is also recorded that how many queries belong to each of the sending node and stored in another buffer whose values are overwritten with every time the quota is calculated.

Then the numbers of different sending nodes are calculated.

After that share of each sending node is calculated by using the formula

Share of each node = [Ni (Queries from ith sender) / Total Queries in Buffer] * Processing Capacity (Q).

We consider the following example to better understand this technique. In this example we have three nodes A, B and C sending queries to node N with the capacity of processing 30 queries (Q) at a time. A sending 100 queries, B sending 25 and C sending 75 queries at a time. According to this query distribution strategy the node A will get the major share of processing resources by applying the formula $\frac{100}{200} \times 30 = 15$ queries similarly B's 4 queries and C's 11 queries will be served. The Fig explains the scenario.

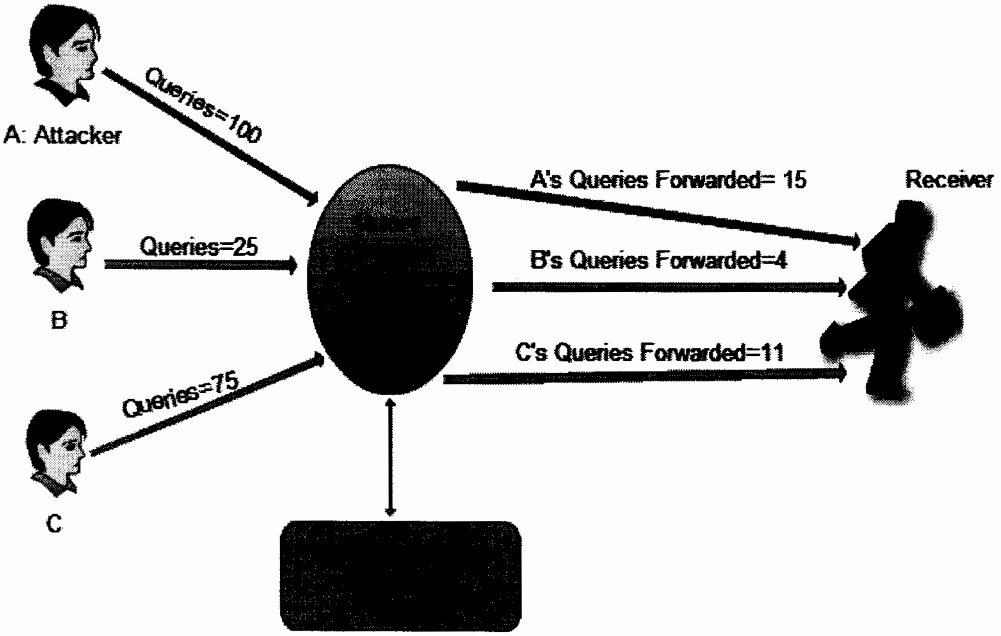

Figure 4.3: Strategy I Weighted QDS with Single Attacker

As we know that a malicious node or the attacker always comes with an intention of sending more and more queries in order to capture maximum resources. Hence in the above example if node A is the single attacker to this network who is sending burst of queries irrespective to the query distribution strategy working at the node N. Then by the applying this strategy node A would receive 50% share of the query processing resources at node N, but yet the remaining 50% of processing resources of receiving node N would be utilized in serving the queries of the legitimate nodes. Malicious queries are also served together with the Good ones. But a single attacker would not be able to contain the whole network and by containing the attack's effect starvation is also minimized. The legitimate work is still being done in light of an attack.

Strategy I:
Weighted Query Distribution Strategy with Multiple attackers:

While considering the same above mentioned example here we have increased the number of attackers to see the working of weighted Query distribution under the impact of multiple attacks. As we know that in blind flooding the adversaries endlessly inject burst of queries to deny the services at the legitimate end.

Here in this example if node A and node C are the attackers, i.e. A sending 100 and C sending burst of 75 malicious queries at a time and node B who is good node sends 25 queries to node N who can serve only 30 queries at a time. Now according to weighted query distribution strategy Node A's 15, node B's 4 and node C's 11 queries would be served. The figure explains the scenario clearly.

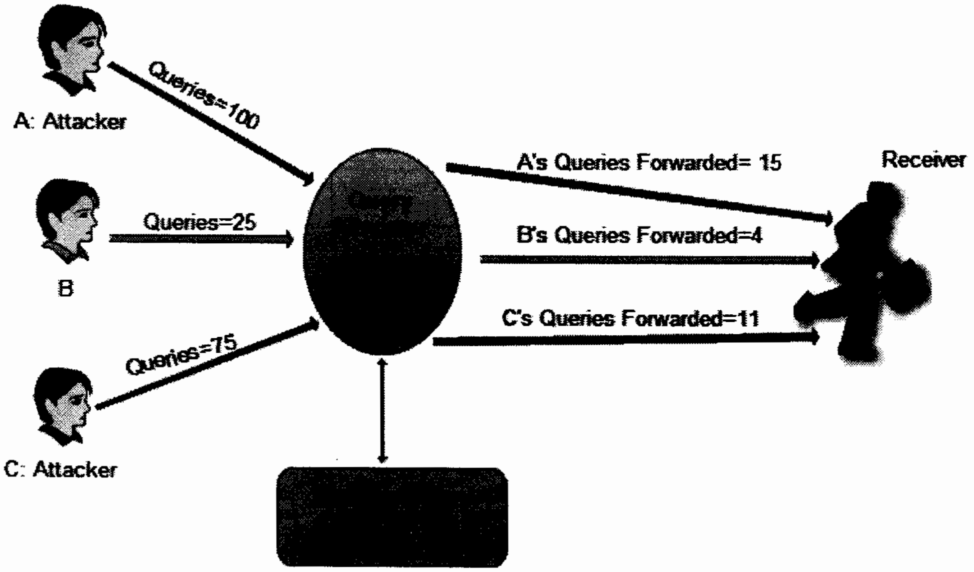

Figure 4.4: Strategy I Weighted QDS with Multiple Attackers

Hence as node A and C are malicious so a total of 26 malicious queries would be served and the good node B's 4 queries would be served. By generalizing it node A would get 50% share of the query processing resources at node N, B would get 13% of the resources for its queries and the node C would get a share of 27% of processing resources. Hence the 87% of the processing resources would be taken by the malicious nodes to serve their queries at node N while yet the remaining 13 % of the processing resources would be utilized to serve the legitimate queries. Hence even the malicious nodes capture the major share of the resources but yet they have been restricted to overwhelm the entire network as the legitimate work is still being carried out with these malicious activities. So a network remains functional even in the light of an attack.

Strategy II:

Fractional Query Distribution Strategy with Single attacker:

In fractional query distribution strategy each incoming link to a node is given the equal fraction of the nodes query processing capacity. For instance if a node has L incoming links to it and Q is the capacity of the receiving node then accordingly this strategy's mathematical formulation is given as below.

Mathematical Calculation of Each Node Share through Fractional QDS:

Each node knows its processing capacity that how much queries it can serve during a particular period of time i.e. Q.

After a particular time each node counts the queries residing in its incoming buffer.

It is also recorded that how many queries belong to each of the sending node and stored in another buffer whose values are overwritten with every time the quota is calculated.

Then the numbers of different sending nodes are calculated. After that share of each sending node is calculated by using the formula.

Share of Each Node= $\frac{Q}{L}$ to each of the nodes for handling there queries. Any of the unused or left over capacity by any of the sending node would go wasted.

An example for understanding, we consider a node N with a capacity of 30 queries, with three links A, B and C i.e. A sending 6 queries, B sending 12, and C sending 10. By applying fractional query distribution strategy each of the node would get a capacity of serving 10 queries from each node. The figure explains the scenario as follows.

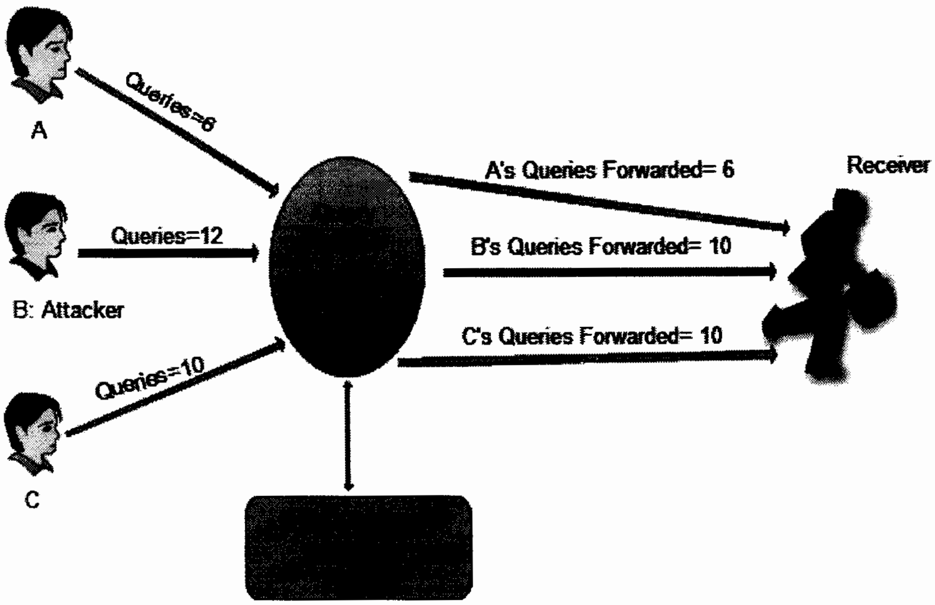

Figure 4.5: Strategy II Fractional QDS with Single Attacker

If B is malicious then off its 12 queries 10 would be served, 2 would be dropped as limit exceeds the allocated capacity. Similarly node A has sent total 6 queries then all of its queries would be served and as its limit is less then the allocated capacity therefore it's leftover capacity would go wasted. C's all the 10 queries would be served and no one would be dropped. Thus by generalizing we can say that 33% of query processing resources would be consumed to serve malicious queries while 53% would be utilized to serve the good queries. Hence we have restricted the attacker from consuming the major share of the resources and serving legitimates nodes during the attack.

Strategy II:

Fractional Query Distribution Strategy with Multiple attackers:

By increasing the number of attackers lets observe the working of fractional query distribution technique under this geometric increase in adversaries. Taking the above example in which for instance we assume that off the three two nodes A and B are malicious. A sending 6, B sending 12 i.e. a total of 18 malicious queries are sent to node N to process while node C is a good node sending 10 queries. And the capacity of node N is 30 queries at a time. Figure explains the scenario.

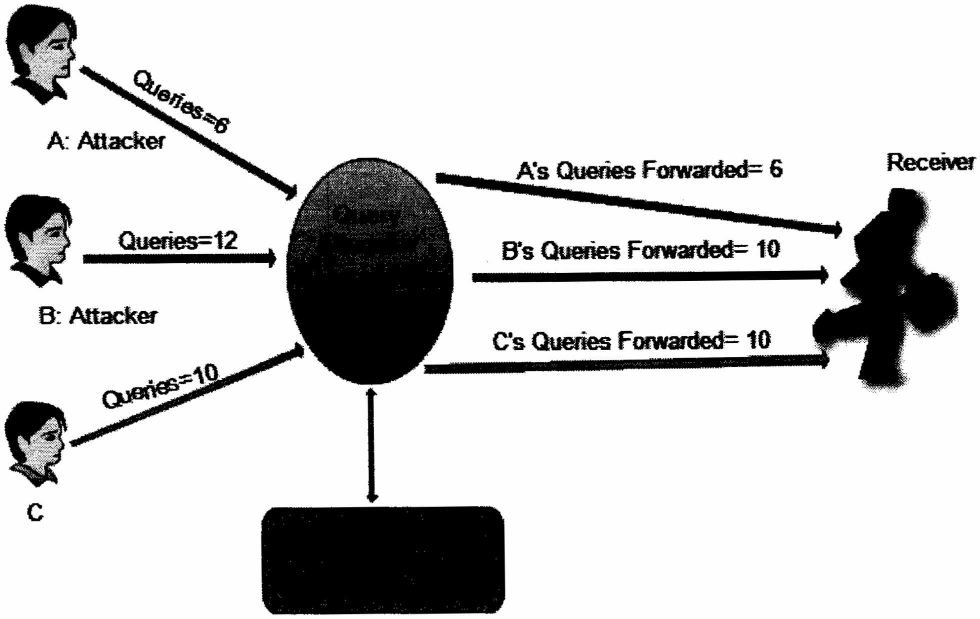

Figure 4.6: Strategy II Fractional QDS with Multiple Attackers

By applying this strategy each nodes 10 queries would be served according to the preset quota. Hence node's A all 6 queries would be served, out of 12 B's 10 and C's all would be served. So a total of 65% of the resources are consumed by malicious queries to be processed while 35 % are consumed to serve the good ones. Although we see that major query processing resources are under the effects of attackers but due to the use of fractional query distribution strategy node is able to continue the legitimate work also.

Strategy III:

Weighted Inverse Query Distribution Strategy with Single attacker:

As the name indicates this strategy is opposite to already discussed weighted strategy. In weighted inverse the one who sends more queries is restricted and allocated the least share of the query processing resources of a node. We believe that it is always the attacker who attempts to maximize its attack by sending heavy burst of queries to overload the node or the network to deny the services at the legitimate end. Through this strategy the effect of this increased multiplicative explosion of malicious queries has been reduced to produce a linear damage to the network's performance.

Mathematical Calculation of Each Node Share through Weighted QDS:

Each node knows its processing capacity that how much queries it can serve during a particular period of time i.e. Q.

After a particular time each node counts the queries residing in its incoming buffer.

It is also recorded that how many queries belong to each of the sending node and stored in another buffer whose values are overwritten with every time the quota is calculated.

Then the numbers of different sending nodes are calculated.

After that share of each sending node is calculated by using the formula as described below.

Formula: Suppose there are L links or nodes who are sending X_1, X_2, X_3, ... X_n queries respectively and the capacity of the receiving node is Q and of the all incoming links Z denotes the link for which is quota to be calculated, then share of each of the node can be calculated as:

Share of node Z = [{1/Queries from node Z (Xz)} / {1/ X_1 + 1/X_2 + 1/X_3, ... 1/X_n }]* Capacity of receiving node (Q) .

Now considering the same example that has been taken in weighted query distribution strategy on this example we would be applying the weighted inverse query distribution strategy to evaluate its performance. As A is the single attacker who is catering an attack by blindly generating excessive amount of queries, it send a burst of 100 queries at a time to receiver node N whose capacity is only 30 queries at a time thus his malicious node generates 70% more traffic then its processing limits. Link B sends 75 and C sends 25 queries to node N. Now according to this strategy node A would get the least share of query processing resources, and node C would be the one whose most of the queries would be served as it is the least sending queries node. As shown in the figure.

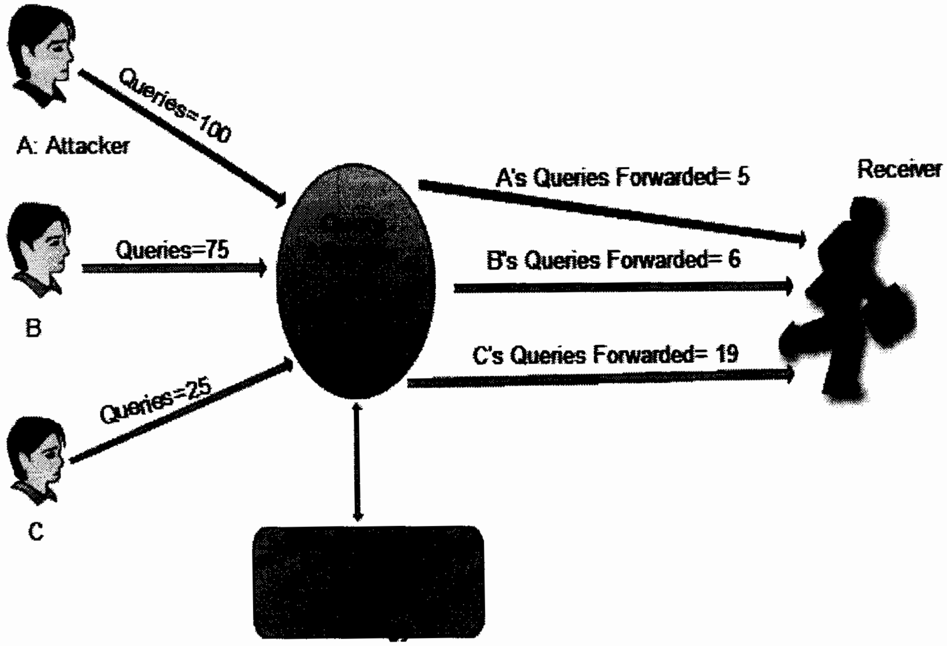

Figure 4.7: Strategy III Weighted Inverse QDS with Single Attacker

Hence A's 5 queries would be served consuming 16% of query processing resources. B's 6 and C's 19 queries would be served together would consume 84% of resources. As it is depicted from above results that with this strategy we can control the attacker's intention of overloading the network by its malicious queries because as there is an increase in mass of queries the lesser the share it will get. And the network is kept functional even under this heavy burst of Query-flood-DoS attack

Strategy III:

Weighted Inverse Query Distribution Strategy with Multiple attackers:

We have simply increased the number of attackers here to see the performance of this strategy under more critical situations. Referencing the same above example, here we simply take both node A and node B as a attacker and node C is the only good node sending 25 queries. A send 100 and B send 75, i.e. both sending heavy amount of queries to maximize their effect as figure explains the scenario best.

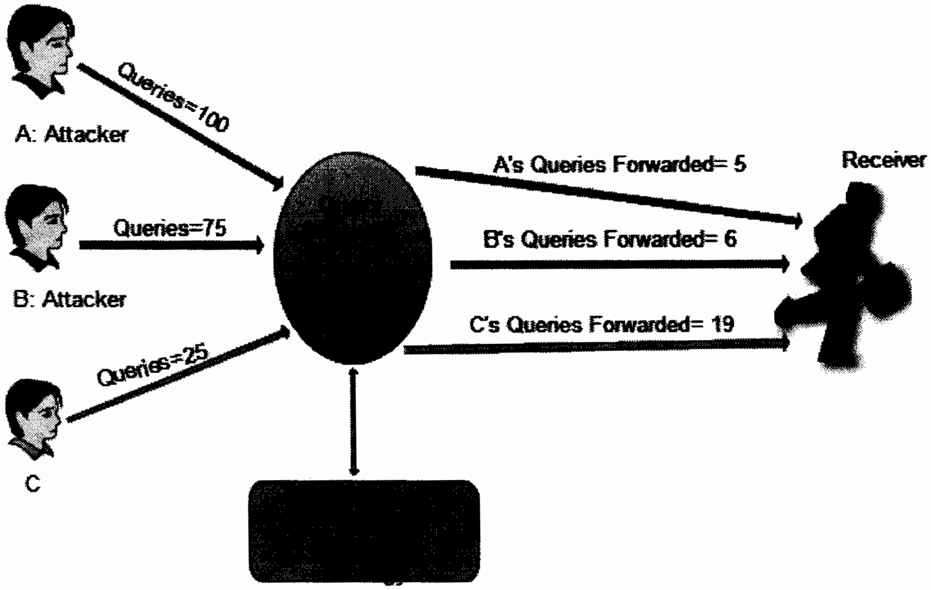

Figure 4.8: Strategy III Weighted Inverse QDS with Multiple Attackers

Thus by applying this strategy both malicious nodes would get a total share of just 36% of resources while the node C which is the good node would get the rest 64% of the processing resources. Hence even under maximum Query flood load by the adversaries the entire network cannot be paralyzed and legitimate users or nodes can not be completely denied from the services.

Strategy IV:

Optimized Fractional Query Distribution Strategy with Single attacker:

It is an extension to the already discussed fractional query distribution strategy. Optimized fractional divides the capacity in the same fashion as the fractional but the difference here is this that the left over capacity at any link who is sending less queries then the allocated quota is spilled over to those links that are injecting extra queries then the assigned capacity.

Mathematical Calculation of Each Node Share through Weighted QDS

Each node knows its processing capacity that how much queries it can serve during a particular period of time.

After a particular time each node counts the queries residing in its incoming buffer.

It is also recorded that how many queries belong to each of the sending node and stored in another buffer whose values are overwritten with every time the quota is calculated.

Then the numbers of different sending nodes are calculated.

After that share of each sending node is calculated by using the formula.

In general

Share of Each node = Capacity of the receiving node (Q) / Total number of nodes sending queries (L)

If (Queries from any node < the allocated quota)

Then

The remaining share will be equally allocated to all other nodes whose total queries are greater then the allocated share.

Taking an example to better understand this strategy. Three nodes A sending 6 queries, B=20 and C= 12 to the node N with capacity of processing 30 queries at a time. Figure explains the scenario.

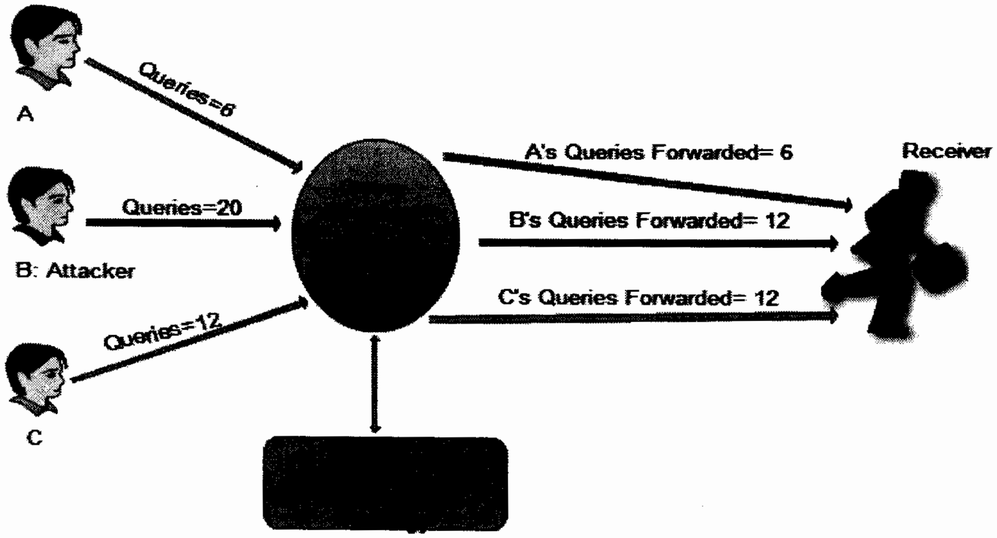

Figure 4.9: Strategy IV Optimized Fractional QDS with Single Attacker

Firstly this strategy divides the capacity fractionally among the three links i.e. from each link 10 queries would be accepted by the receiving node. Hence A's all 6, B's 10 and C's also 10 queries would be served, the left over capacity at node A is not wasted it is again used an is allocated to the rest of the nodes with larger number of queries i.e. B's and C's two more queries would be served each. Now off B's 20 queries 12 would be served and C's all the 12 would be served. If lets say B is the attacker it would consume 40% of the resources rest are consumed in serving legitimate queries, so we remain successful with optimized fractional strategy in distributing the resources fairly and restricting single attacker's control over the entire network.

Strategy IV:

Optimized Fractional Query Distribution Strategy with Multiple attackers:

With an increased number of attackers we would be analyzing this strategy. Taking five nodes A, B, C, D and E sending traffic to node N having capacity of processing 30 queries at a time. For instance A, B, and C are attackers. A sending 300 queries B= 200 and C=500, while D=2 and E=15. As shown in the figure.

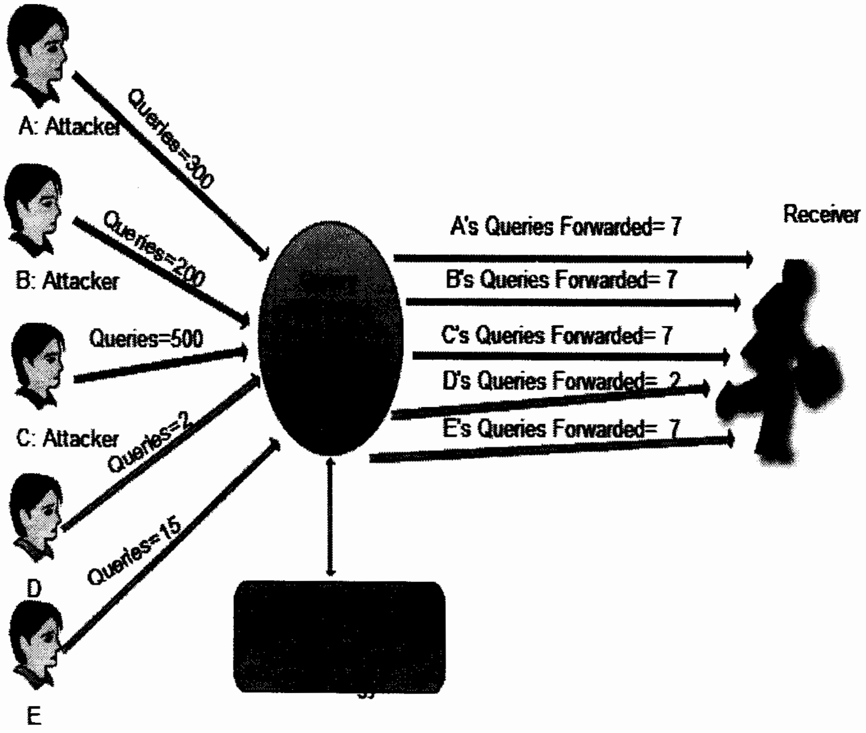

Figure 4.10: Strategy IV Optimized Fractional QDS with Multiple Attacker

According to the applied strategy each of the nodes would receive a link quota of processing 6 queries at a time. As the node D has some left over capacity it is spilled over to rest of the nodes with more queries so A, B, C and E's 7 queries would be served each. We can see that off the 1000 malicious queries by the adversaries only 21 would be served and 9 good queries. Even under the heavy multiple attacker's effects the network would be prevented from overwhelming and it remains functional.

Strategy V:

Complete Trusted Query Distribution Strategy with Single attacker:

Prior to the working of this strategy it is assumed that before exchanging any query the nodes have established a trust relationship between each other. The complete trust strategy first of all would serve all the queries of the trusted nodes, and then the un-trusted would be served only in the case if any capacity is left over.

Mathematical Calculation of Each Node Share through Weighted QDS:

Each node knows its processing capacity that how much queries it can serve during a particular period of time.

After a particular time each node counts the queries residing in its incoming buffer.

It is also recorded that how many queries belong to each of the sending node and stored in another buffer whose values are overwritten with every time the quota is calculated.

Then the numbers of different sending nodes are calculated.

After that share of each sending node is calculated by using the general formula as described below.

By fractionally dividing the capacity among all the trusted node i.e. $\frac{Q}{T}$ (where T= number of trusted nodes) and after serving all the trusted nodes completely yet if there is some left over capacity i.e. if total queries of Trusted nodes$< Q$ then it is spilled over among un trusted nodes fractionally

i.e. $\frac{Q\text{−queries of }T}{\text{no of un trusted nodes}}$. But if total queries of T node $\geq Q$ then quota of un trusted nodes = 0.

Considering an example in which we have two trusted nodes A and B while C is a single attacker three of them sending traffic to the receiver node N with the capacity of 30 queries at a time. A sends 25, B sends 20 and C sends 50 queries at a time figure shows the scenario.

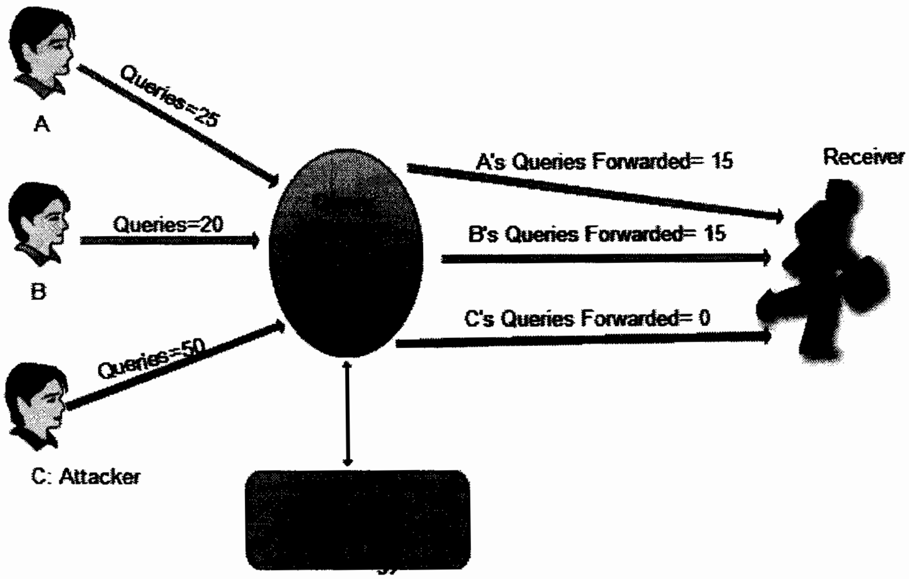

Figure 4.11 Strategy V Complete Trusted QDS with Single Attacker

According to the applied strategy first of all the queries of the trusted nodes would be served. Quota would be allocated to these trusted nodes fractionally, i.e. each of these first 15 queries would be allocated the resources. So A's 15 and B's 15 queries would be served as the capacity is fully consumed in fulfilling the requests of trusted nodes only so no capacity is left to serve the node C which is malicious. Hence 100% resources are consumed to serve trusted nodes queries and the malicious node fails to conduct any damage to the network. Hence the network resources are prevented from wastage and all the legitimate queries are served.

Strategy V:

Complete Trusted Query Distribution Strategy with Multiple attackers:

Under more complex situation we would analyze the performance of this strategy. Taking an example of six nodes A, B, C, D, E and F. off these A and B are trusted nodes while

C, D and E are malicious where as E is a good node. A sending 10, B=25, C=50, D=100, E=200 and F=20 queries to node N with the capacity of processing 30 queries at a time. On applying this Strategy first the trusted nodes requests would be served. Scenario is explained with the help of the figure below.

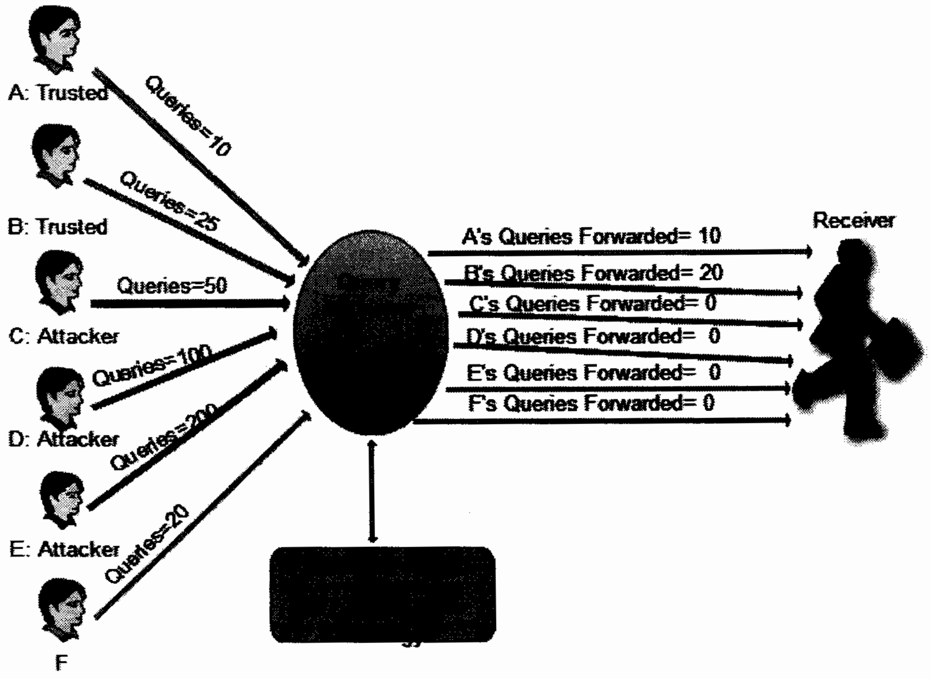

Figure 4.12: Strategy V Complete Trusted QDS with Multiple Attackers

Hence by dividing the Quota A's all 10 would be served and it's left quota will be allocated to the rest of the trusted nodes first so B's now 20 queries would be served and no quota is left for the other nodes queries. Therefore the malicious nodes C, D and E are blocked to cause any damage but together with them the Good nodes like F are also deprived off from their share of resources, so some legitimate nodes would also starve. An enhancement to this strategy is double trust distribution strategy explained as follows.

Strategy VI:

Double Trusted Query Distribution Strategy with Single attacker:

This strategy is an extension to above mentioned strategy; it is also based on the same assumption as above. According to this strategy the quota is fractionally divided among all the incoming links. Then of the incoming links some would be trusted nodes and some would be un- trusted nodes. Then the quota allocated to both the categories of the node are different, trusted nodes are always given the priority over other nodes. i.e. trusted node given double the share of the un-trusted nodes.

Mathematical Calculation of Each Node Share through Weighted QDS:

Each node knows its processing capacity that how much queries it can serve during a particular period of time.

After a particular time each node counts the queries residing in its incoming buffer.

It is also recorded that how many queries belong to each of the sending node and stored in another buffer whose values are overwritten with every time the quota is calculated.

Then the numbers of different sending nodes are calculated.

After that share of each sending node is calculated by using the formula as below.

If 'L' is the total number of links or nodes to node N and 'U' is the number of un trusted nodes and X is representing the link whose quota is to be set in a time, then the quota of nodes would be set according to $X = \frac{Q}{U+2(L-U)}$.

If a node is a trusted node then its share will be 2X and If a node is a un-trusted node then its share will be X

We consider an example here to have a better understanding of the strategy. We take six nodes A, B, C, D, E and F sending queries to node N with the capacity of processing 64 queries at a time. Off the six nodes A and B are trusted and rest four are un-trusted off the un-trusted D is malicious. A sending 20 queries, B=20, C=10, D=40, E=10 and F sends 20 queries. According to applied strategy un-trusted nodes 8 queries would be served each while trusted would get the double share so A and B 16 queries would be served each. Figure shows the scenario.

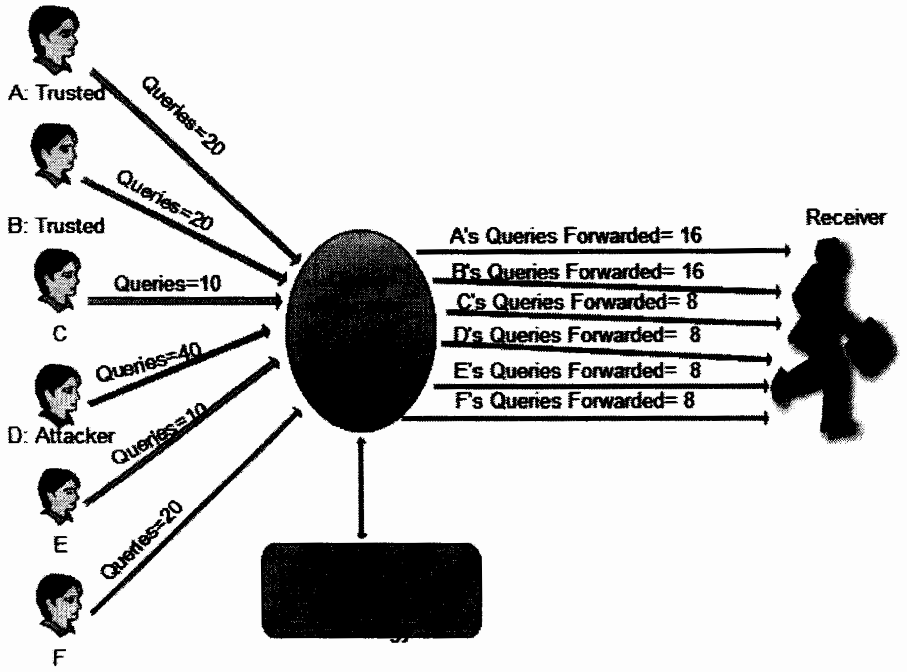

Figure 4.13: Strategy VI Double Trusted QDS with Single Attacker

Hence with this Double trust query distribution not only the trusted nodes are always served but together with them the rest of the nodes also get their share of resources. We can see then malicious node also gets its share but only 12% of resources are consumed to serve its queries 88% are consumed by the legitimate work. Thus we are successful in giving a fair share of resources to each of the nodes. A single attacker can no longer overwhelm the whole network by its malicious queries.

Strategy VI:

Double Trusted Query Distribution Strategy with Multiple attackers:

Here we have simply increased the number of attackers to test the performance of this strategy under such complex situation. Taking the above example and increasing the number of adversaries in it. A, B are trusted and now off the un-trusted C, D and E are malicious. The capacity of the node is 64 queries at a time as shown in the figure.

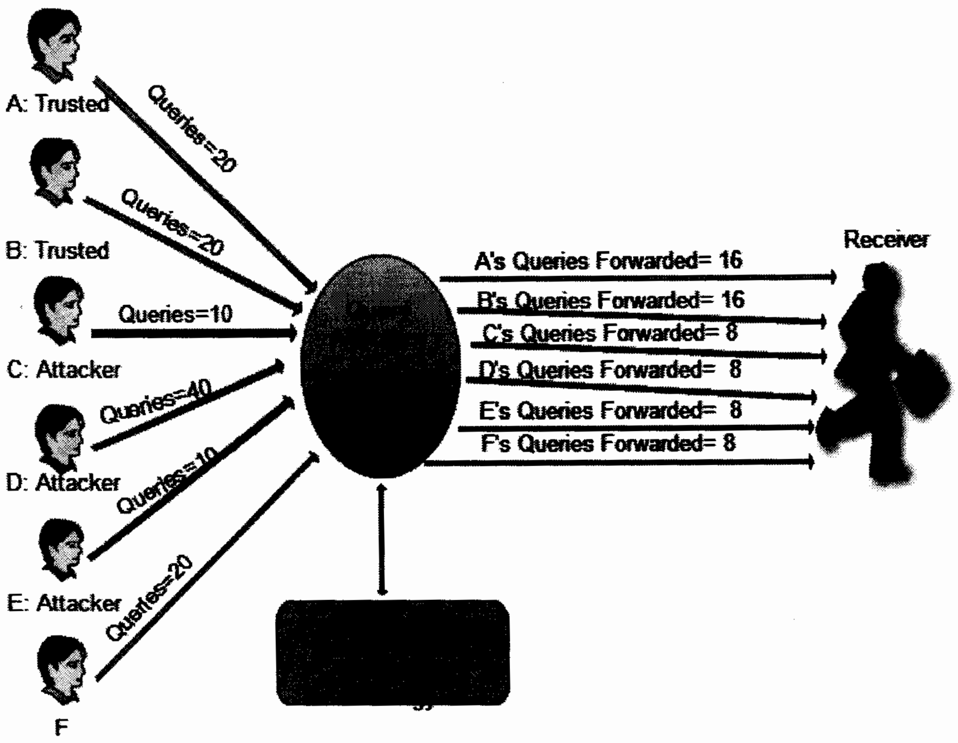

Figure 4.14: Strategy VI Double Trusted QDS with Multiple Attackers

We see that with the increased malicious nodes even the network is functional as trusted nodes are given priority. And this strategy fairly distributes resources among every incoming link. No matter how many queries are injected by these malicious nodes, these can not bring massive damage to the network because adversaries are unable to paralyze the entire network.

4.3 Chapter Summary:

In this chapter we have provided the solution to cater with the application layer query flood DoS attack in MANETs. We have developed here different Query Distribution Strategies (QDS) which not only fairly distribute the resources among the nodes in the network, but also mitigate the malicious effects during the attack time. Thus these strategies help in reducing the starvation at the legitimate end. These are based on simple mathematical models which do not require any complex processing capabilities and consume less energy and power for operation. All the strategies are built while considering the both types of flooding in mind. So that the variant performance of these strategies can be observed and measured.

5. Results

5 Results:

We have evaluated these different strategies by using Network Simulator 2. Simulations are run by using different parameters with respect to changing scenarios to measure the performance of these strategies and obtain the results. Here before deriving the results a brief introduction of the simulator used is presented first. So that the reader has the ease in understanding the working of the tool, which would be a first hand help for proper understanding of results.

5.1 About Network Simulator-2:

NS-2 is a simulator used to build network simulations. We have used it to build network and then observing the traffic to analyze the performance of the QDR's under different scenarios. It is an Object oriented simulator which uses C++ at the backend and at the frontend it uses OTcl interpreter. This simulator supports a class of hierarchy in both ends; in C++ it is known as Compiled Hierarchy while in OTcl it is known as Interpreted Hierarchy. Both the hierarchies have one-to-one correspondence with each other i.e. one compiled hierarchy with one interpreter hierarchy according to user's perspective. The Class TclObject is the root of this hierarchy. Through the Interpreter the new simulator objects are created by the users. These objects are initialized in the interpreter then they are matched with the relevant object in the compiled hierarchy. The pre-defined methods in the class Tclclass automatically create the interpreted class hierarchy. The user defined objects are mirrored or matched to the pre-defined methods in the class TclObject.

NS-2 has to perform two different types of tasks so it uses two separate languages one for each task. For simulations of protocols it requires a system programming language which is capable of working with bytes, packets headers and capable of using such algorithms that can accommodate heavy set of data. Run-time is more valuable than turn-around time for such type of tasks. On the other hand there are such network simulations for which the iteration time or turn-around time is more important; these simulations involve such scenarios which require quick analysis of configurations or some parameters. NS-2 provides us a platform which can accommodate both the needs i.e. we can run simulations for the real networks and also can carry out the analysis for different scenarios and different parameters.

For these two reasons Ns-2 uses two languages C++ and OTcl. C++ is appropriate for the detailed implementation of the protocol as it is fast enough to run but to changes it is slower. Where as OTcl is slower in running but it is adaptive to rapid changes and is interactive thus this makes it suitable for simulation configuration.

Yet confusion arises in understanding why two languages are used. In order to remove this confusion it is briefly explained here where to use which language. OTcl is best to be used for one-time work, for configuration or for setting up the different parameters like delay, queuing etc. But in order to work with some new parameters which were not supported by OTcl then one has to use C++ as it would allow creating new objects, it also allows to make changes to the behavior of the existing C++ class. NS-2 has many classes defined in it, off them the six important classes are as follows.

- Embedded Tcl
- Inst Var
- Tcl
- Tcl Object
- Tcl Class
- Tcl Command

5.2 Reasons to use NS-2:

There are many reasons to use Ns-2 simulator than any other, off them few are as follows.

- ✓ It is a freeware.
- ✓ Its help is easily available at the internet.
- ✓ Many user manual guides are there, which facilitate in understanding and working with simulator.
- ✓ Its interface is quite interactive then any other simulator.

5.3 Results Validation:

To justify our proposed solution and to validate its effectiveness we conducted different number of experiments. We evaluated Remote work, energy consumption and fairness among the proposed query distribution strategies.

To calculate the results we take an ad-hoc network of 30 nodes. All the calculations are performed on the receiver. Routing is performed through AODV. Every node sends 50 to 300 queries per second. The maximum number of queries that a node can send are 300. The receiving node can serve maximum 1000 queries per second so the capacity is 1000. The simulation is run for 100 seconds. Number of malicious nodes varies from 0 to 10. X dimension of the topography is 1600; Y dimension of the topography is 1600. MAC protocol is 802.11 and bandwidth is 4 Mbps. Size of the each packet is 100 bytes. In case of blind flooding the number of queries from a malicious node is 300.

5.4 Remote Work:

Remote work is the work done by a specific node on behalf of other nodes. A node can consume its resources either to serve its own queries or those of remote nodes. Execution of remote node queries is called as remote work.

We measured the results of each of the strategy by taking two cases-Blind Flooding and Intelligent Flooding.

5.4.1 Legitimate and Malicious Remote Work by using Query Distribution Strategies-Blind Flooding:

We calculated legitimate and malicious remote work performed in case of each query distribution strategies. Every malicious node sends as many query as possible to the 16th node. Since maximum 300 queries per second are allowed to send by a single node so every malicious node sends 300 queries to the 16th node. First results are calculated by applying Weighted Query Distribution Strategy (QDS). Figure 5.1 is showing the results.

Pseudo Code for Getting Results For Graph 1-6:

1-We Conducted 11 experiments by changing the values of number of malicious nodes from 0 to 10 and the Blind-flooding strategy has been used while conducting these experiments.

2- For each experiment we:

I. Calculate total queries sent by all the senders.

II. After that we calculate that how many queries are served from each of the sender by using the specified query distribution technique (This value is calculated by the formula of specified QDS).

III. Then we measure that what was the status of the each of the node (Malicious or Legitimate (Status of each of the user is defined at the start of the experiment)).

IV. Then we calculate that how many queries of the legitimate nodes are served out of the total served queries.

V. Then we measured how many queries of the malicious nodes are served out of the total served queries.

3- We plot the graph between values of step IV and V of all the 11 experiments.

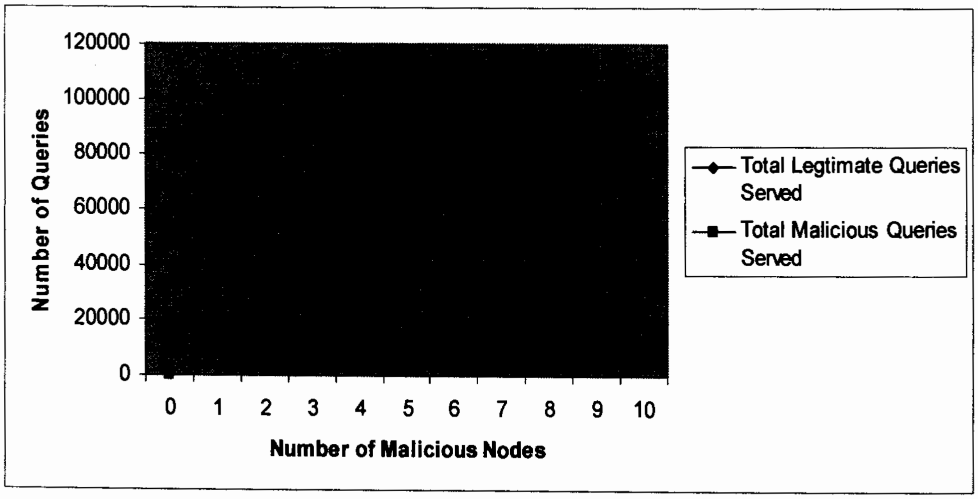

Figure 5.1: Legitimate and Malicious Remote Work in case of Weighted QDS

It is clear from the figure 5.1 that when there is no malicious node the legitimate remote work is 100% (100,000 queries processed within 100 seconds) and malicious remote work is 0. When the number of malicious nodes starts to increase the legitimate remote work begins to decrease and the malicious remote work commences to increase. When out of 15 sending nodes 10 were malicious the legitimate remote work was 18% and malicious remote work was 82%.

After that results are calculated by applying Fractional Query Distribution Strategy (QDS). Figure 5.2 is showing the results.

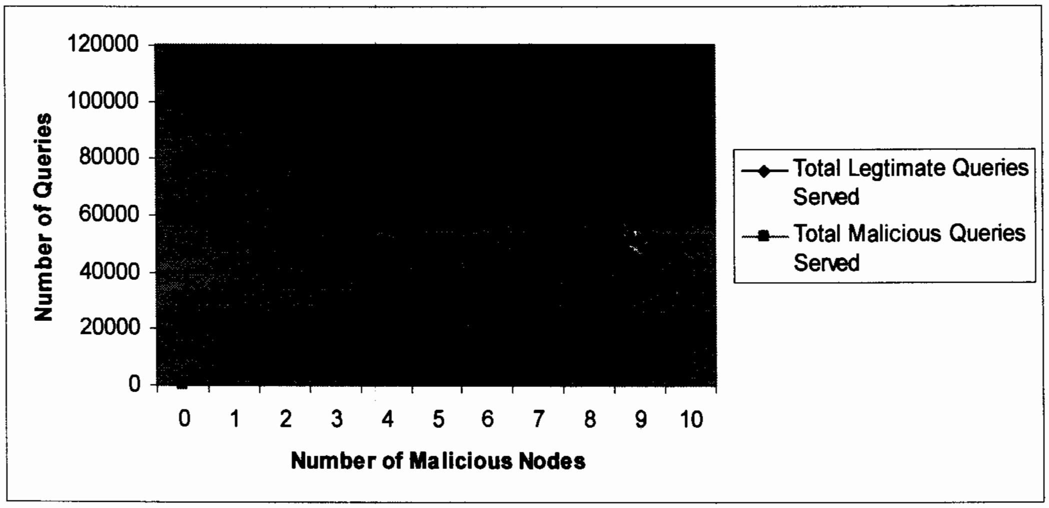

Figure 5.2: Legitimate and Malicious Remote Work in case of Fractional QDS

Figure 5.2 is stating that when there is no malicious node the legitimate remote work is 100% (100,000 queries processed within 100 seconds) and malicious remote work is 0. When the number of malicious nodes starts to increase the legitimate remote work begins to decrease and the malicious remote work commences to increase. When out of 15 sending nodes 10 were malicious the legitimate remote work was 33% and malicious remote work was 67%.

In the third instance we calculated the results by applying Weighted Inverse Query Distribution Strategy (QDS). Figure 5.3 is showing the results.

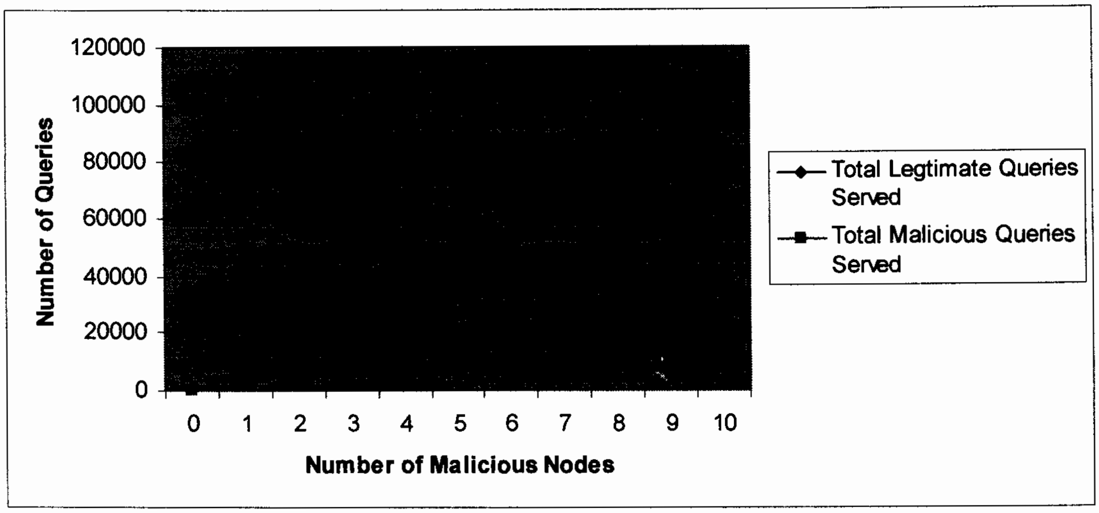

Figure 5.3: Legitimate and Malicious Remote Work in case of Weighted Inverse QDS

Figure 5.3 is stating that when there is no malicious node the legitimate remote work is 100% (100,000 queries processed within 100 seconds) and malicious remote work is 0. When the number of malicious nodes starts to increase the legitimate remote work begins to decrease and the malicious remote work commences to increase. When out of 15 sending nodes 10 were malicious the legitimate remote work was 59% and malicious remote work was 41%.

In the fourth experiment we calculated the results by applying Optimized Fractional Query Distribution Strategy (QDS). Figure 5.4 is showing the results.

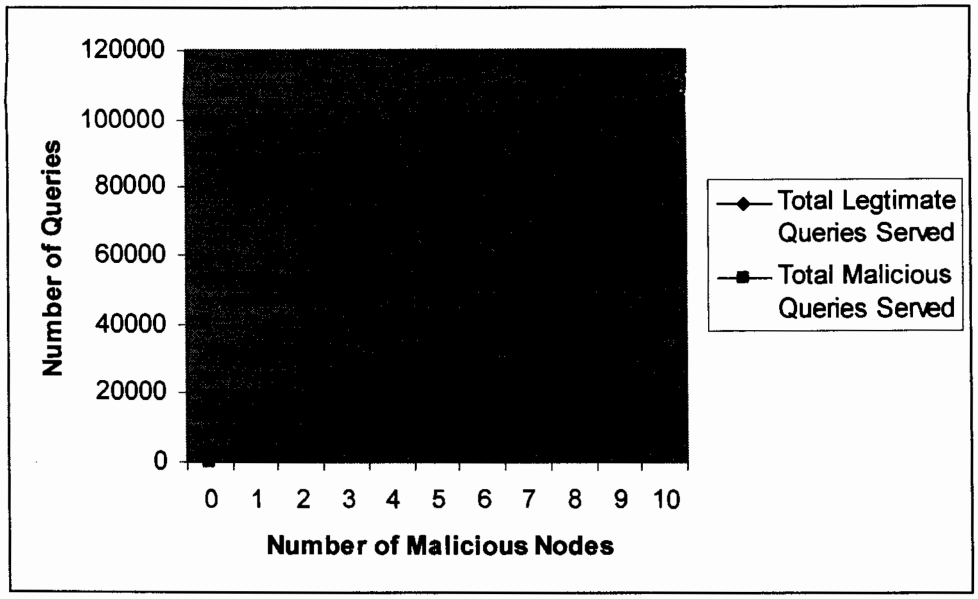

Figure 5.4: Legitimate and Malicious Remote Work in case of Optimized Fractional QDS

Figure 5.4 is stating that when there is no malicious node the legitimate remote work is 100% (100,000 queries processed within 100 seconds) and malicious remote work is 0. When the number of malicious nodes starts to increase the legitimate remote work begins to decrease and the malicious remote work commences to increase. When out of 15 sending nodes 10 were malicious the legitimate remote work was 59% and malicious remote work was 41%.

In the fifth experiment we calculated the results by applying Double Trusted Query Distribution Strategy (QDS). Figure 5.5 is showing the results.

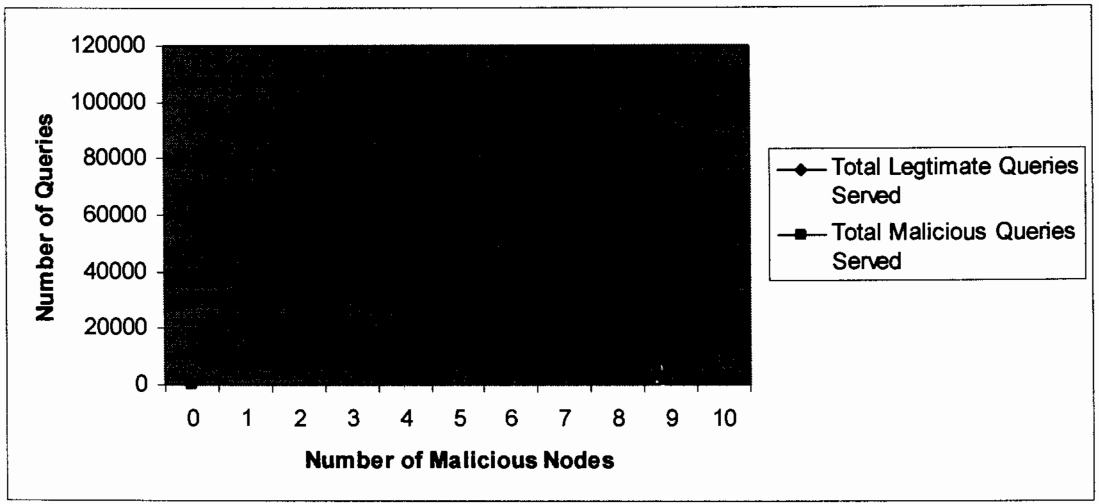

Figure 5.5: Legitimate and Malicious Remote Work in case of Double Trusted QDS

Figure 5.5 is stating that when there is no malicious node the legitimate remote work is 100% (100,000 queries processed within 100 seconds) and malicious remote work is 0. When the number of malicious nodes starts to increase the legitimate remote work begins to decrease and the malicious remote work commences to increase. When out of 15 sending nodes 10 were malicious the legitimate remote work was 42% and malicious remote work was 58%.

In the sixth experiment we calculated the results by applying Complete Trusted Query Distribution Strategy (QDS). Figure 5.6 is showing the results.

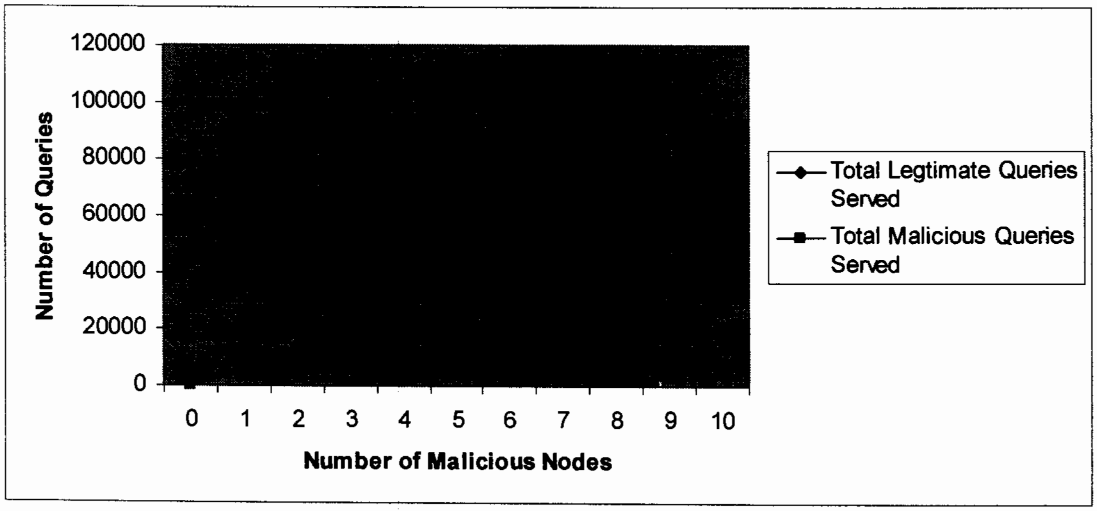

Figure 5.6: Legitimate and Malicious Remote Work in case of Complete Trusted QDS

Figure 5.6 is stating that when there is no malicious node the legitimate remote work is 100% (100,000 queries processed within 100 seconds) and malicious remote work is 0. When the number of malicious nodes starts to increase the legitimate remote work begins to decrease and the malicious remote work commences to increase. When out of 15 sending nodes 10 were malicious the legitimate remote work was 52% and malicious remote work was 48%.

5.4.2 Legitimate and Malicious Remote Work by using Query Distribution Strategies-Intelligent Flooding:

When intelligent flooding is applied 5 from the 6 proposed strategies deliver the same results as it was in case of blind flooding. Because these 5 strategies require to send maximum number of queries to get more share. That is the reason why these strategies continue to flood as many queries as possible. However one of the strategies (Weighted Inverse) reveals different results as compared to that of blind flooding. Figure 5.7 is explaining the results.

Pseudo Code for Getting Results for Graph 7:

1-We Conducted 11 experiments by changing the values of number of malicious nodes from 0 to 10 and changing the flooding strategy to intelligent and observed the functionality of this QDS.

2- For each experiment we:

VI. Calculate total queries sent by all the senders.

VII. After that we calculate that how many queries are served from each of the sender by using the Weighted Inverse query distribution technique (This value is calculated by the formula of this QDS).

VIII. Then we measure that what was the status of the each of the node (Malicious or Legitimate (Status of each of the user is defined at the start of the experiment)).

IX. Then we calculate that how many queries of the legitimate nodes are served out of the total served queries.

X. Then we measured how many queries of the malicious nodes are served out of the total served queries

3- We plot the graph between values of step IV and V of all the 11 experiments.

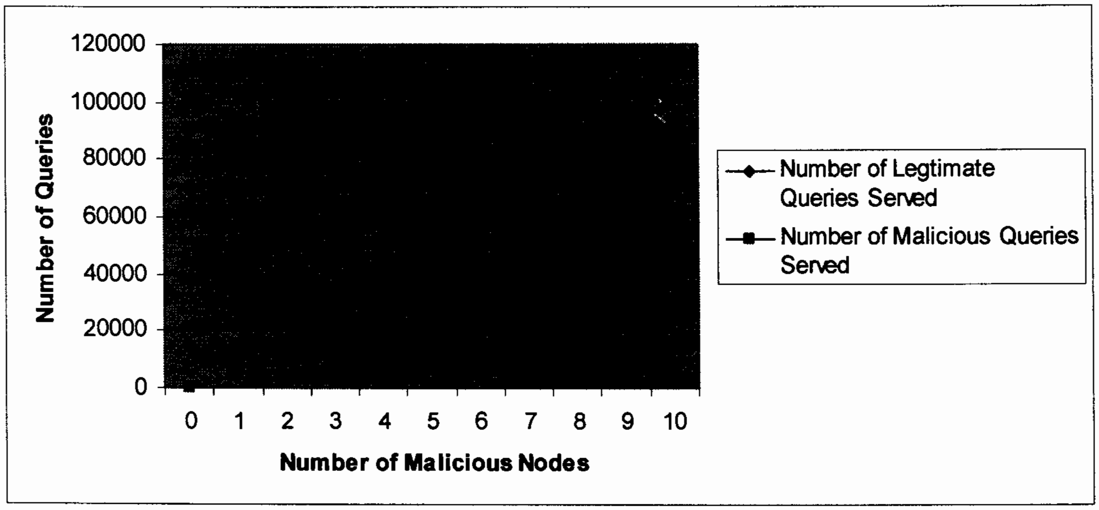

Figure 5.7: Legitimate and Malicious Remote Work in case of Weighted Inverse QDS-Intelligent Flooding

Figure 5.7 is stating that when there is no malicious node the legitimate remote work is 100% (100,000 queries processed within 100 seconds) and malicious remote work is 0. When the number of malicious nodes starts to increase the legitimate remote work begins to decrease and the malicious remote work commences to increase. When out of 15 sending nodes 10 were malicious the legitimate remote work was 50% and malicious remote work was 50%. Figure 5.8 is explaining the legitimate and malicious remote work done by using blind and intelligent flooding while the underlying strategy was Weighted Inverse QDS.

Pseudo Code for Getting Results of Graph 5.8:

```
1- For each type of flooding (Blind and Intelligent)

2-We Conducted 11 experiments by changing the values of number of malicious nodes from 0 to 10

3- For each experiment we:

    3.1- Calculate total queries sent by all the senders.

    3.2- After that we calculate that how many queries are served from each of the sender by using the Weighted query distribution technique (This value is calculated by the specified formula of this QDS).

    3.3- Then we measure that what was the status of the each of the node (Malicious or Legitimate (Status of each of the user is defined at the start of the experiment)).

    3.4- Then we calculate that how many queries of the legitimate nodes are served out of the total served queries.

    3.5- Then we measured how many queries of the malicious nodes are served out of the total served queries.

4- We plot the graph between values of step 3.4 and 3.5 of all the 11 experiments for each type of flooding
```

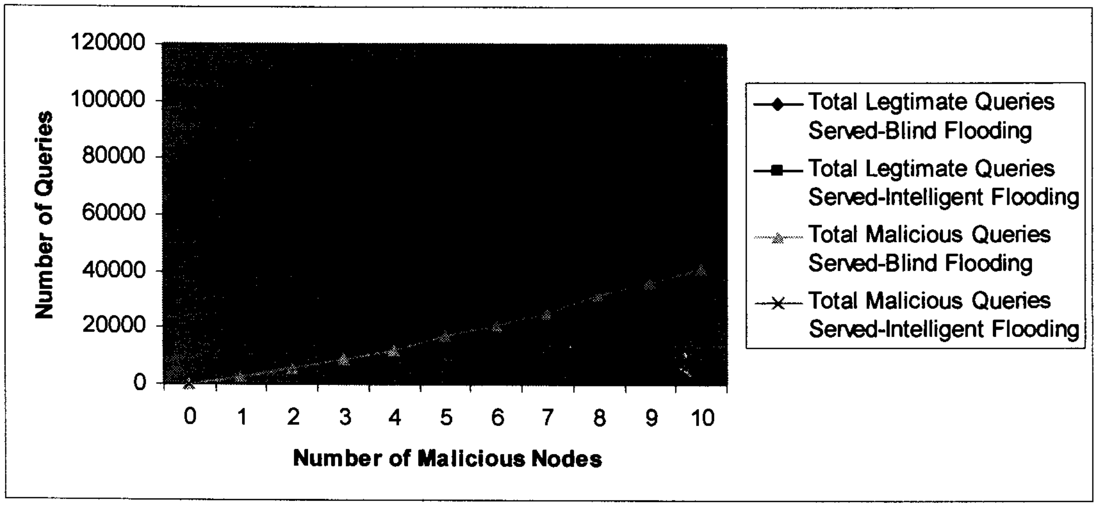

Figure 5.8: Legitimate and Malicious Remote Work in case of Blind and Intelligent Flooding-Weighted QDS

5.4.3 Total Queries Sent by Malicious Node in Blind and Intelligent Flooding:

Five from the six proposed QDSs performs same amount of legitimate and malicious remote work in case of blind and intelligent flooding. But the total number of queries sent by the malicious nodes is different in three of the proposed QDS (Fractional, Weighted Inverse, and Optimized Fractional) when the underlying flooding strategy is changed from blind to intelligent. Figure 5.9 is explaining the total number of queries sent and served by malicious node in case of blind flooding and intelligent flooding when underlying strategy was Fractional QDS.

Pseudo Code for Getting Results of Graph 5.9:

```
1- For each type of flooding (Blind and Intelligent).

2-We Conducted 11 experiments by changing the values of number of malicious nodes from 0 to 10.

3- For each experiment we:

    3.1- Calculate total queries sent by all the senders.

    3.2- After that we calculate that how many queries were malicious from the total received queries.

    3.3- After that we calculated that how many queries are served from each of the sender by using the Fractional query distribution technique (This value is calculated by the specific formula of this QDS).

    3.4- Then we measure that what was the status of the each of the node (Malicious or Legitimate (Status of each of the user is defined at the start of the experiment)).

    3.5- We calculated the total malicious queries sent while Fractional QDS is implemented wit respect to both flooding strategies and compared the results. Then we measured how many queries of the malicious nodes are served out of the total sent queries.

4- We plot the graph between values of step 3.2 and 3.5 of all the 11 experiments for each type of flooding.
```

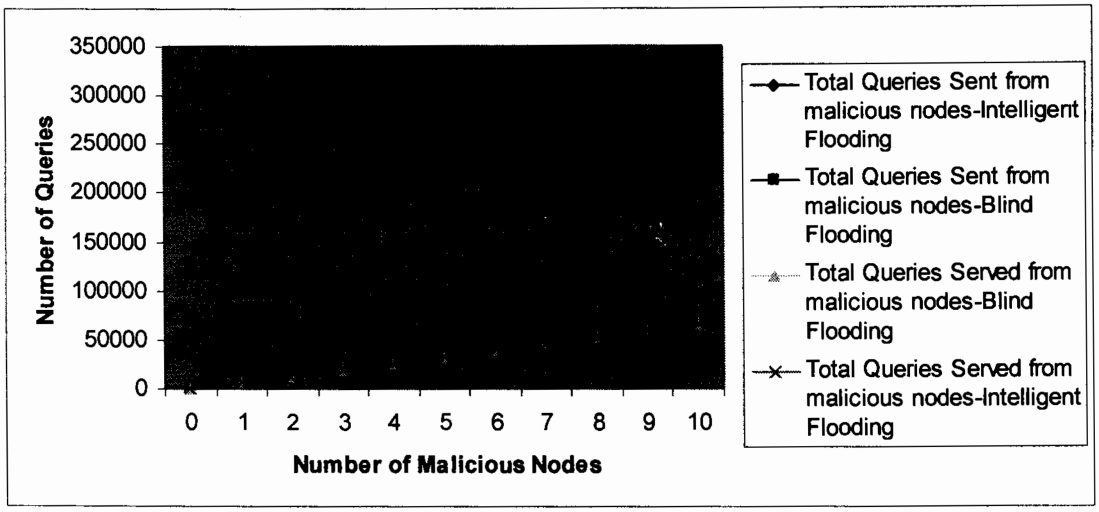

Figure 5.9: Total Queries Sent by Malicious Nodes-Fractional QDS

From the figure 5.9 we can see that in case of blind flooding 10 malicious nodes sent up to 300,000 queries to get 67% resources of a node. Sending of 300,000 queries itself requires too much power and resources at the attacker end. So in case of blind flooding malicious nodes are consuming too many resources to attack. While in case of intelligent flooding since the malicious nodes have knowledge about the underlying query distribution strategy so it sends requests accordingly. So in case of intelligent flooding malicious nodes send 67000 queries to occupy 67% of the victim's resources. So the total queries sent by malicious nodes in case of intelligent flooding are almost 4.5 times lower as compared to that of blind flooding.

In the next experiment we measured the total queries sent by malicious nodes in case of blind and intelligent flooding when the underlying strategy was Weighted Inverse QDS. Figure 5.10 is summarizing the results.

Pseudo Code for Getting Results of Graph 5.10:

```
1- For each type of flooding (Blind and Intelligent).

2-We Conducted 11 experiments by changing the values of number of malicious nodes from 0 to 10.

3- For each experiment we:

    3.1- Calculate total queries sent by all the senders.

    3.2- After that we calculate that how many queries were malicious from the total received queries.

    3.3- After that we calculated that how many queries are served from each of the sender by using the Weighted Inverse query distribution technique (This value is calculated by the specific formula of this QDS).

    3.4- Then we measure that what was the status of the each of the node (Malicious or Legitimate (Status of each of the user is defined at the start of the experiment)).

    3.5- We calculated the total malicious queries sent while Weighted QDS is implemented wit respect to both flooding strategies and compared the results regarding the resources cosumed. Then we measured how many queries of the malicious nodes are served out of the total sent queries.

4- We plot the graph between values of step 3.2 and 3.5 of all the 11 experiments for each type of flooding.
```

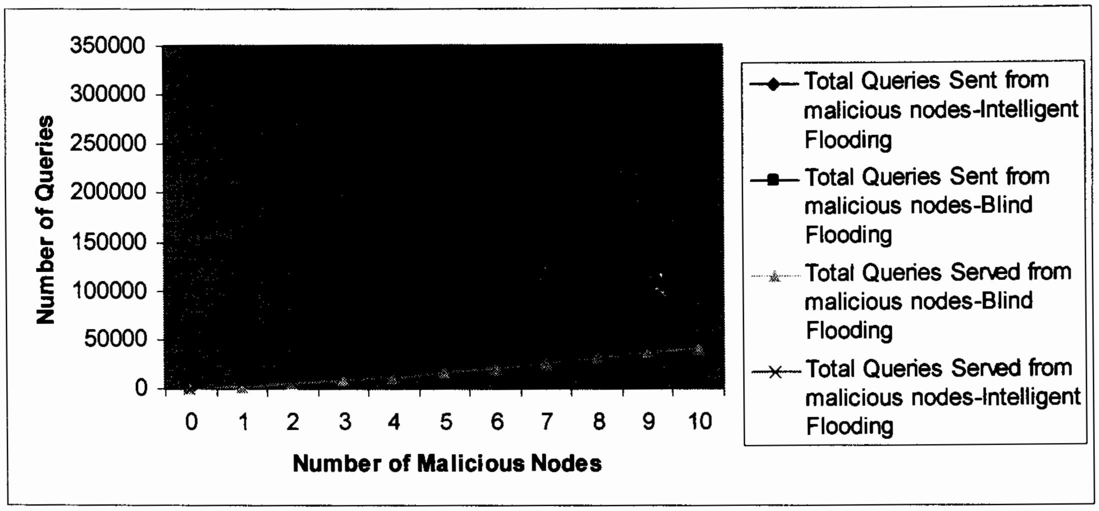

Figure 5.10: Total Queries Sent by Malicious Nodes-Weighted Inverse QDS

From the figure 5.10 we can see that in case of blind flooding 10 malicious nodes sent 300,000 queries to get 42% resources of a node. In case of intelligent flooding malicious nodes send 50,000 queries to occupy 50% of the victim's resources. So the total queries sent by malicious nodes in case of intelligent flooding are almost 6 times lower as compared to that of blind flooding and the resources occupied in case of intelligent flooding are 8% higher.

5.4.4 Comparison of Legitimate and Malicious Remote Work in case of Blind Flooding:

In the first experiment we measured and compared the legitimate work done by each of the six proposed QDS in case of blind flooding. Results are summarized in figure 5.11.

Pseudo Code for Getting Results of Graph 5.11:

```
1-For each of the query distribution strategy we:
    1.1- Conducted 11 experiments by changing the values of number of malicious
    nodes from 0 to 10 under Blind-Flooding Strategy.
    1.2- For each experiment we:
        1.2.1-We calculate total queries sent by all the senders.
```

1.2.2- After that we calculate that how many queries are served from each of the sender by applying all the six QDS one by one (Depends upon QDS).

1.2.3- Then we measure that what was the status of the each of the node (Malicious or Legitimate).

1.2.4- Then we calculate that how many queries of the legitimate nodes are served out of the total served queries to calculate the Legitimate remote Work.

1.3- Then we draw a graph for the values of 1.2.4 of that particular QDS.

2- Final graph consists of values of all the 6 QDS.

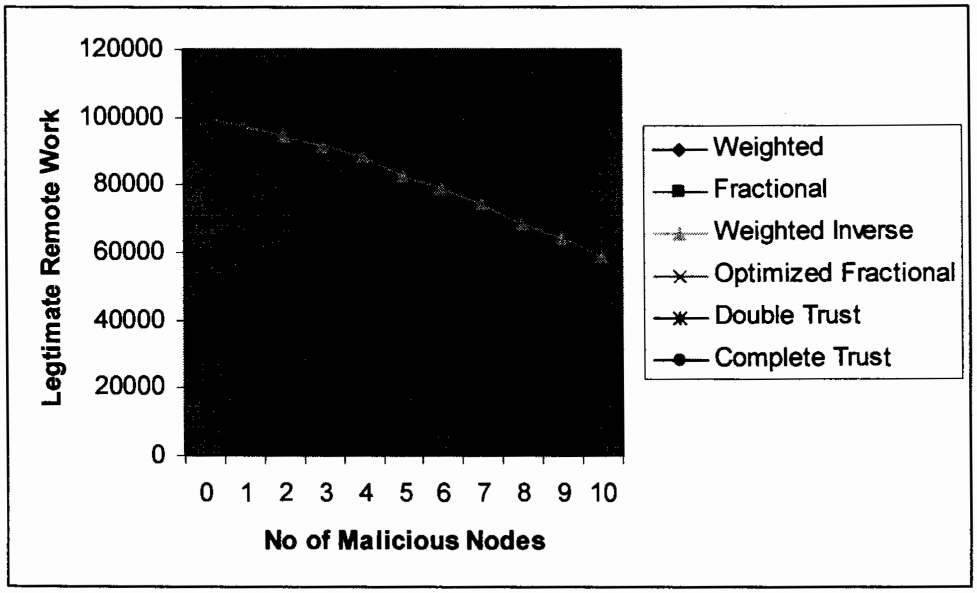

Figure 5.11: Comparison of Legitimate Remote Work-Blind Flooding

It can easily conclude from the figure 5.11 that Weighted Inverse strategy is delivering more as compared to other five strategies. Complete Trusted is second after the weighted inverse while Weighted QDS is performing very poor and at the last position.

In the second experiment we measured and compared the malicious work done by each of the six proposed QDS in case of blind flooding. Results are summarized in figure 5.12.

Pseudo Code for Getting Results of Graph 5.12:

```
1-For each of the query distribution strategy we:

    1.1- Conducted 11 experiments by changing the values of number of malicious
    nodes from 0 to 10 under Blind-Flooding Strategy.

    1.2- For each experiment we:

        1.2.1-We calculate total queries sent by all the senders.

        1.2.2- After that we calculate that how many queries are served from each
        of the sender by applying all the six QDS one by one (Depends upon
        QDS).

        1.2.3- Then we measure that what was the status of the each of the node
        (Malicious or Legitimate).

        1.2.4- Then we calculate that how many queries of the malicious nodes are
        served out of the total served queries to calculate the Malicious remote
        Work.

    1.3- Then we draw a graph for the values of 1.2.4 of that particular QDS.

2- Final graph consists of values of all the 6 QDS.
```

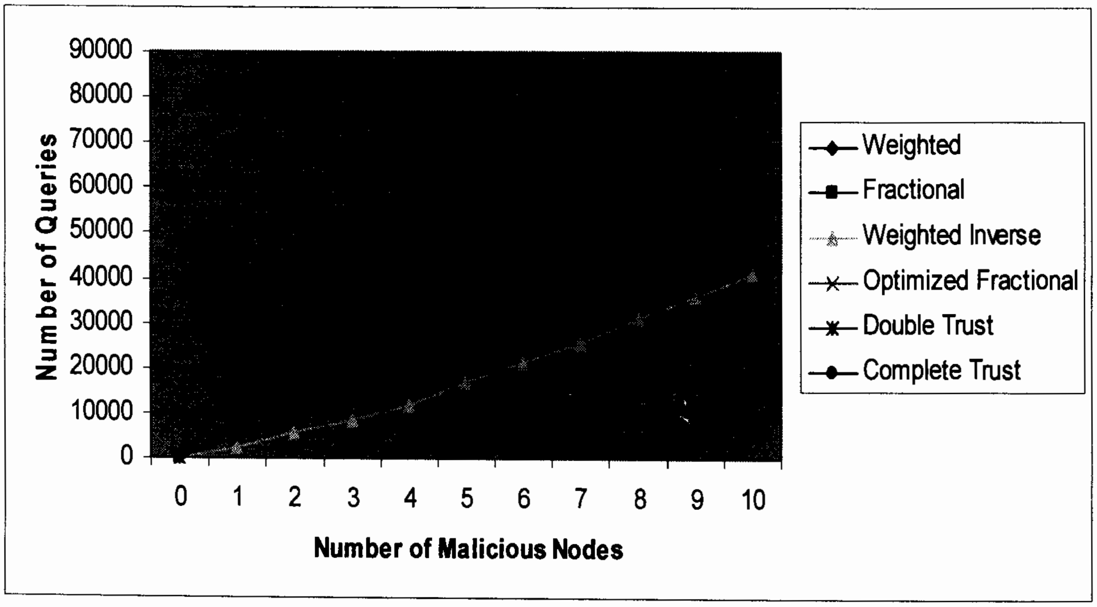

Figure 5.12: Comparison of Malicious Remote Work-Blind Flooding

In Weighted QDS malicious work is maximum and it is more vulnerable to attack. Weighted Inverse and Complete Trusted QDSs are performing better because malicious work is very low in these two strategies.

5.4.5 Comparison of Legitimate and Malicious Remote Work in case of Intelligent Flooding:

In the first experiment we measured and compared the legitimate work done by each of the six proposed QDS in case of intelligent flooding. Results are summarized in figure 5.13.

Pseudo Code for Getting Results of Graph 5.13:

```
1-For each of the query distribution strategy we:

    1.1- Conducted 11 experiments by changing the values of number of malicious
    nodes from 0 to 10 under Intelligent-Flooding Strategy.

    1.2- For each experiment we:

        1.2.1-We calculate total queries sent by all the senders.

        1.2.2- After that we calculate that how many queries are served from each
        of the sender by applying all the six QDS one by one (Depends upon
        QDS).

        1.2.3- Then we measure that what was the status of the each of the node
        (Malicious or Legitimate).

        1.2.4- Then we calculate that how many queries of the legitimate nodes
        are served out of the total served queries to calculate the Legitimate
        remote Work.

    1.3- Then we draw a graph for the values of 1.2.4 of that particular QDS.

2- Final graph consists of values of all the 6 QDS.
```

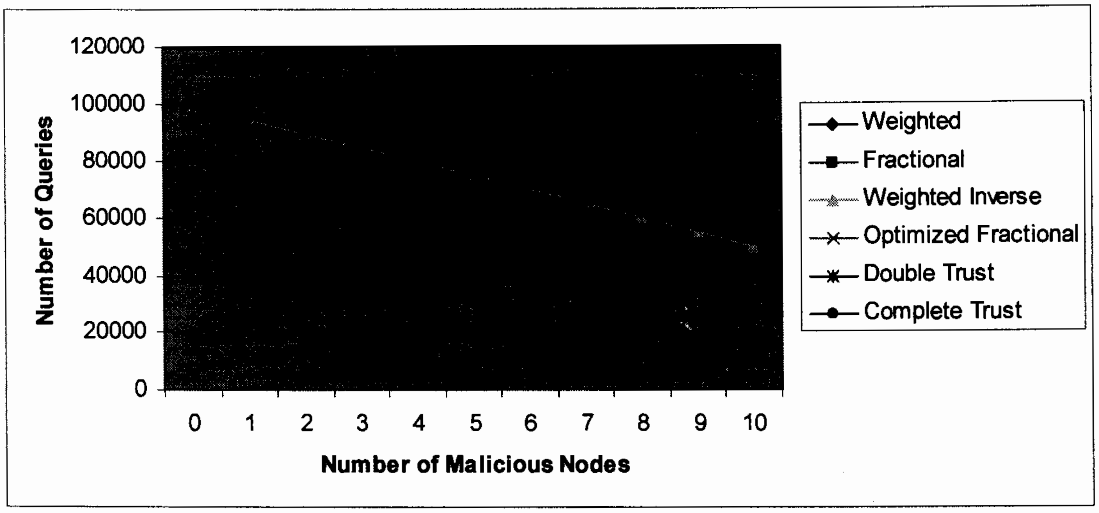

Figure 5.13: Comparison of Legitimate Remote Work-Intelligent Flooding

When the proposed six QDSs are compared in by applying intelligent flooding it is measured that Complete Trusted QDS becomes the best choice. The weighted inverse moves to number 2 and Weighted is still at the last number.

In the second experiment we measured and compared the malicious work done by each of the six proposed QDS in case of intelligent flooding. Results are summarized in figure 5.14.

Pseudo Code for Getting Results of Graph 5.14:

```
1-For each of the query distribution strategy we:
    1.1- Conducted 11 experiments by changing the values of number of malicious
    nodes from 0 to 10 under Intelligent-Flooding Strategy.
    1.2- For each experiment we:
        1.2.1-We calculate total queries sent by all the senders.
        1.2.2- After that we calculate that how many queries are served from each
        of the sender by applying all the six QDS one by one (Depends upon
        QDS).
        1.2.3- Then we measure that what was the status of the each of the node
        (Malicious or Legitimate).
```

1.2.4- Then we calculate that how many queries of the Malicious nodes are served out of the total served queries to calculate the Malicious remote Work.

1.3- Then we draw a graph for the values of 1.2.4 of that particular QDS.

2- Final graph consists of values of all the 6 QDS.

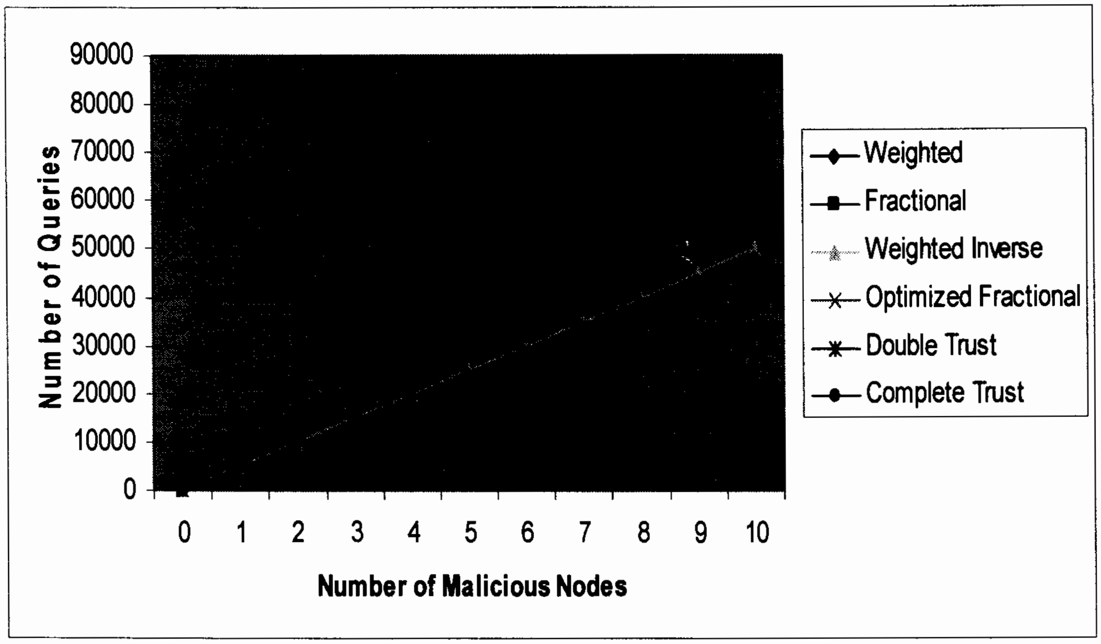

Figure 5.14: Comparison of Malicious Remote Work-Intelligent Flooding

In Weighted QDS malicious work is maximum and it is more vulnerable to attack. Complete Trusted and Weighted Inverse QDSs are performing better because malicious work is very low in these two strategies.

5.4.6 Percentage of Total Legitimate Queries Served from the Total Legitimate Queries Received –Blind Flooding:

In this experiment we measured that how many legitimate queries are served from the legitimate nodes. From the figure 5.15 it can be concluded that the complete trust strategy serves more percentage of legitimate queries as compared to the other five strategies.

Weighted strategy is at the last and serves very few legitimate queries from the total legitimate queries received.

Pseudo Code for Getting Results of Graph 5.15:

```
1- For each of the query distribution strategy

    1.1- We conducted 11 experiments by changing number of malicious nodes and
         for each experiment under Blind-Flooding.

         1.1.1- We check the status of each node (Malicious or Legitimate).
         1.1.2- We calculate the total queries received from all the nodes whose
         status was Legitimate.
         1.1.3- Then queries are served according to the applied strategy.
         1.1.4- After that we calculate how many queries are served from the nodes
         whose status was Legitimate.
         1.1.5- We calculate the percentage from the values of step 1.1.4 and 1.1.2.
    1.2- We add values in the graph calculated in step 1.1.5 for all experiments.
2- We plot graph for the values of all the QDS.
```

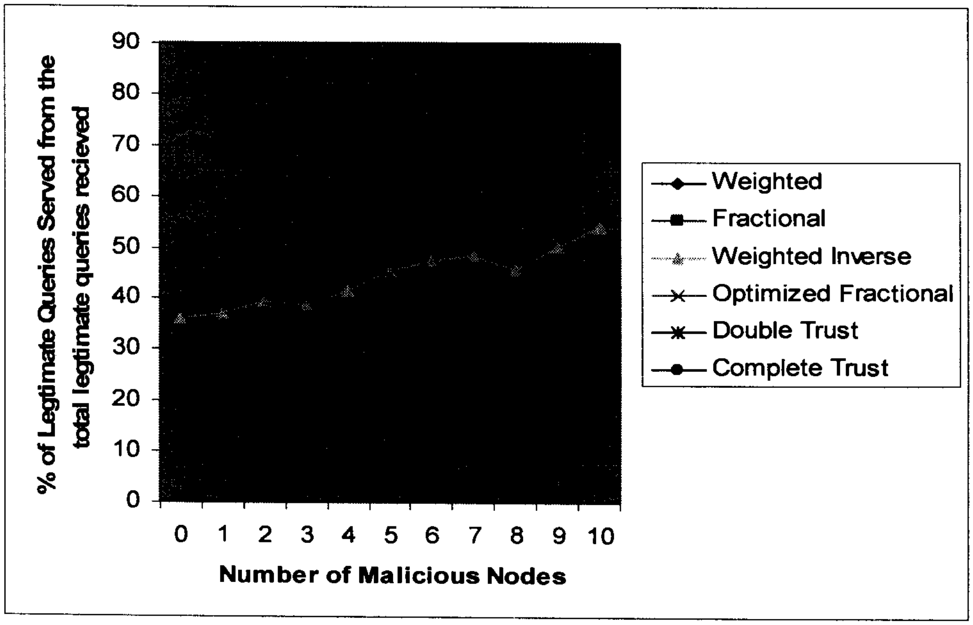

Figure 5.15: Percentage of Total Legitimate Queries Served from the Total Legitimate Queries Received – Blind Flooding

5.4.7 Percentage of Total Malicious Queries Served from the Total Malicious Queries Received –Blind Flooding:

In this experiment it is measured that what percentage of malicious queries is served from the total malicious queries received. Figure 5.16 is showing that Weighted strategy serves the maximum number of malicious packets and weighted inverse serves the minimum number of malicious packets. Complete Trusted strategy also serves very few malicious queries and is on the 2nd last number in figure 5.16.

Pseudo Code for Getting Results of Graph 5.16:

```
1- For each of the query distribution strategy
        1.1- We conducted 11 experiments by changing number of malicious nodes and
             for each experiment under Blind-Flooding.
             1.1.1- We check the status of each node (Malicious or Legitimate).
             1.1.2- We calculate the total queries received from all the nodes whose
             status was Malicious.
             1.1.3- Then queries are served according to the applied strategy.
             1.1.4- After that we calculate how many queries are served from the nodes
             whose status was Malicious.
             1.1.5- We calculate the percentage from the values of step 1.1.4 and 1.1.2.
        1.2- We add values in the graph calculated in step 1.1.5 for all experiments.
2- We plot graph for the values of all the QDS.
```

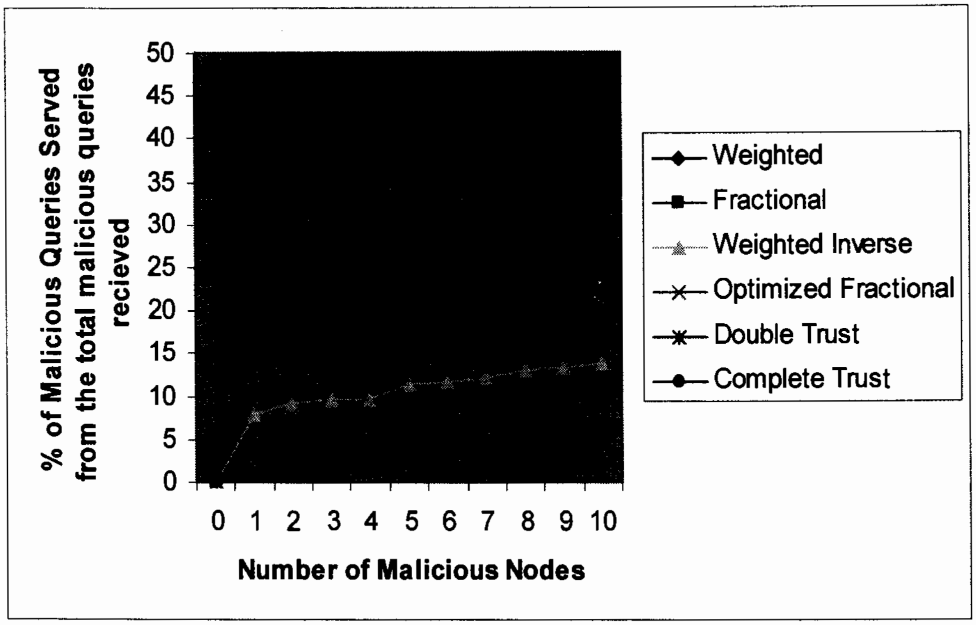

Figure 5.16: Percentage of Total Malicious Queries Served from the Total Malicious Queries Received – Blind Flooding

5.4.8 Percentage of Total Legitimate Queries Served from the Total Legitimate Queries Received –Intelligent Flooding:

The experiment of 5.1.6 is repeated by applying the intelligent flooding technique and it is found that by applying the intelligent flooding technique still complete trust strategy is serving the maximum number of legitimate queries while weighted and weighted inverse strategies are at the bottom and serve very few legitimate queries from the total received as shown in the figure 5.17.

Pseudo Code for Getting Results of Graph 5.17:

```
1- For each of the query distribution strategy

        1.1- We conducted 11 experiments by changing number of malicious nodes and
             for each experiment under Intelligent-Flooding.

             1.1.1- We check the status of each node (Malicious or Legitimate).
             1.1.2- We calculate the total queries received from all the nodes whose
             status was Legitimate.
             1.1.3- Then queries are served according to the applied strategy.
```

1.1.4- After that we calculate how many queries are served from the nodes whose status was Legitimate.

1.1.5- We calculate the percentage from the values of step 1.1.4 and 1.1.2.

1.2- We add values in the graph calculated in step 1.1.5 for all experiments.

2- We plot graph for the values of all the QDS.

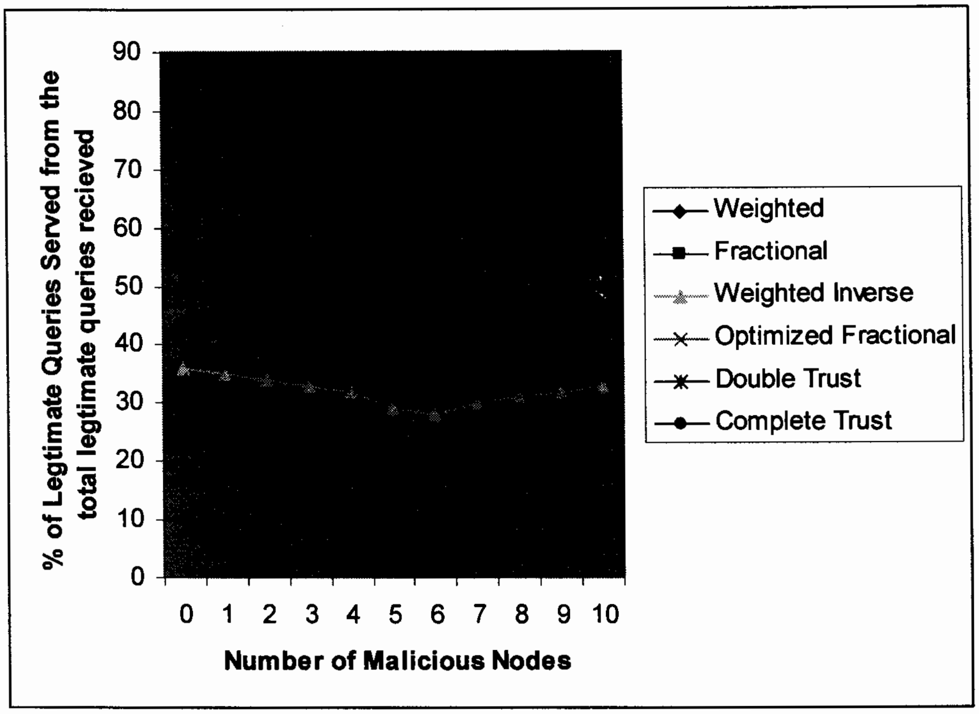

Figure 5.17: Percentage of Total Legitimate Queries Served from the Total Legitimate Queries Received – Intelligent Flooding

5.4.9 Percentage of Total Malicious Queries Served from the Total Malicious Queries Received –Intelligent Flooding:

This experiment gives very different results as compared to the above three experiments. Since the intelligent flooding is applied so the total malicious queries sent by the malicious nodes are very little in case of fractional, optimized fractional and weighted inverse strategy and maximum of the sent queries are served. The reason of high percentage of malicious query service in that the malicious nodes send the queries according to the strategy applied. The results are summarized in figure 5.18.

Pseudo Code for Getting Results of Graph 5.18:

```
1- For each of the query distribution strategy
        1.1- We conducted 11 experiments by changing number of malicious nodes and
             for each experiment under Intelligent-Flooding.
             1.1.1- We check the status of each node (Malicious or Legitimate).
             1.1.2- We calculate the total queries received from all the nodes whose
             status was Malicious.
             1.1.3- Then queries are served according to the applied strategy.
             1.1.4- After that we calculate how many queries are served from the nodes
             whose status was Malicious.
             1.1.5- We calculate the percentage from the values of step 1.1.4 and 1.1.2.
        1.2- We add values in the graph calculated in step 1.1.5 for all experiments.
2- We plot graph for the values of all the QDS.
```

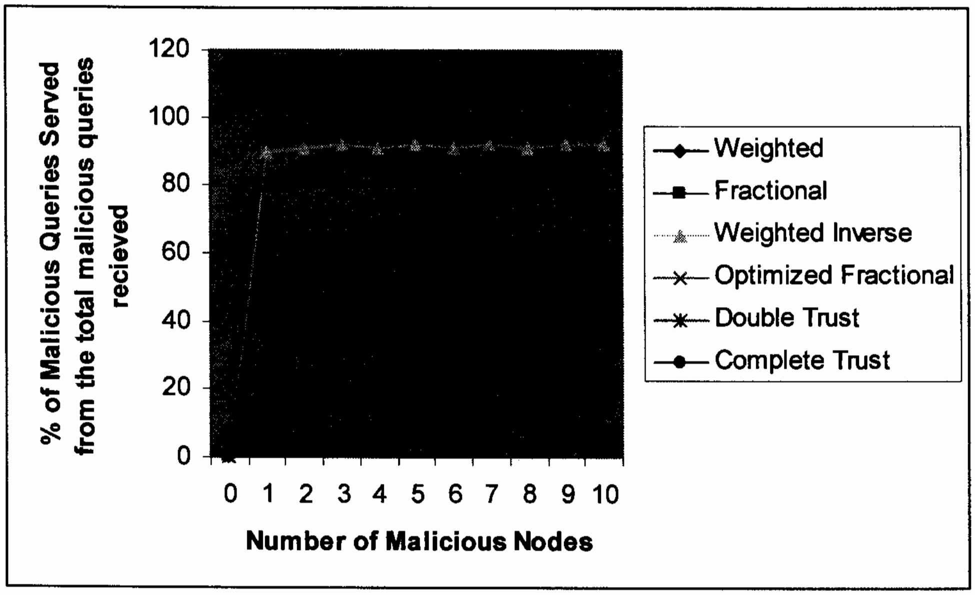

Figure 5.18: Percentage of Total Malicious Queries Served from the Total Malicious Queries Received – Intelligent Flooding

5.5 Energy:

Since nodes of the ad hoc network are battery operated so energy is one of the main performance parameter that should be considered while making any solution. After calculating the remote work we conducted number of experiments to calculate the consumption of energy by the proposed strategies. When there was no malicious node the total consumed energy was about 280 joules but with the addition of malicious nodes the total consumed energy starts to increase and when there was 10 malicious nodes the total consumed energy was 500 joules. With the addition of malicious nodes the portion of energy consumed by the total queries and served queries of legitimate nodes starts to decrease and that of malicious nodes starts to increase. When there was no malicious 280 joules all are consumed by the legitimate node but when there were 10 malicious nodes the total legitimate energy consumed was just 100 joules while energy consumed by the malicious nodes was 400 joules. The above mentioned figures are same for all the remaining strategies. The difference is in case of served queries.

5.5.1 Energy Consumption in Weighted QDS:

The energy calculations are shown in the figure 5.19. In case of zero malicious nodes energy consumed to serve legitimate queries was 132 joules. In the scenario when out of total 15 nodes 10 nodes were malicious the energy consumed by the served queries of legitimate nodes was 24 joules and energy consumed by the served queries of malicious nodes was 108 joules.

Pseudo Code for Getting Results of Graph 5.19:

```
1-We Conducted 11 experiments by changing the values of number of malicious nodes
from 0 to 10

2- For each experiment we:
    2.1- Calculate the initial amount of energy that the node has
    2.2- Calculate total queries sent by all the senders.
    2.3- Calculate that how many queries are served from each of the sender by using
    the Weighted query distribution technique (the value is calculated from the
    formula of this QDS).
    2.4- Queries are served according to the calculations of step 2.3
```

2.5- Calculate the current remaining energy
2.6- Check the status of each of the node (Malicious or Legitimate)
2.7- Calculate the number of legitimate queries served and number of malicious queries served
2.8- Calculate the difference between values of step 2.1and 2.5
2.9- Divide the value of step 2.6 into two categories (Malicious and Legitimate) on the basis of number of queries served from legitimate node and malicious node

3- We plot the graph between values of step 2.9 of all the 11 experiments.

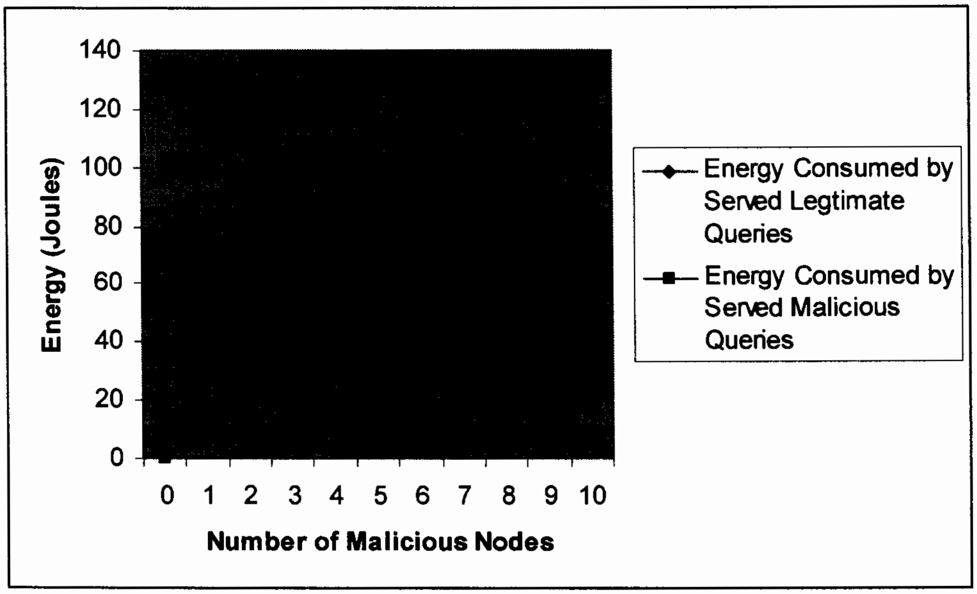

Figure 5.19: Energy Consumption by Served Queries-Weighted QDS

5.5.2 Energy Consumption in Fractional QDS:

The energy calculations are shown in the figure 5.20. In case of zero malicious nodes energy consumed to serve legitimate queries was 132 joules. In the scenario when out of total 15 nodes 10 nodes were malicious the energy consumed by the served queries of legitimate nodes was 44 joules and energy consumed by the served queries of malicious nodes was 88 joules.

Pseudo Code for Getting Results of Graph 5.20:

1-We Conducted 11 experiments by changing the values of number of malicious nodes from 0 to 10.

2- For each experiment we:

2.1- Calculate the initial amount of energy that the node has

2.2- Calculate total queries sent by all the senders.

2.3- Calculate that how many queries are served from each of the sender by using the Fractional query distribution technique (the value is calculated from the formula of this QDS).

2.4- Queries are served according to the calculations of step 2.3.

2.5- Calculate the current remaining energy.

2.6- Check the status of each of the node (Malicious or Legitimate).

2.7- Calculate the number of legitimate queries served and number of malicious queries served.

2.8- Calculate the difference between values of step 2.1and 2.5.

2.9- Divide the value of step 2.6 into two categories (Malicious and Legitimate) on the basis of number of queries served from legitimate node and malicious node

3- We plot the graph between values of step 2.9 of all the 11 experiments.

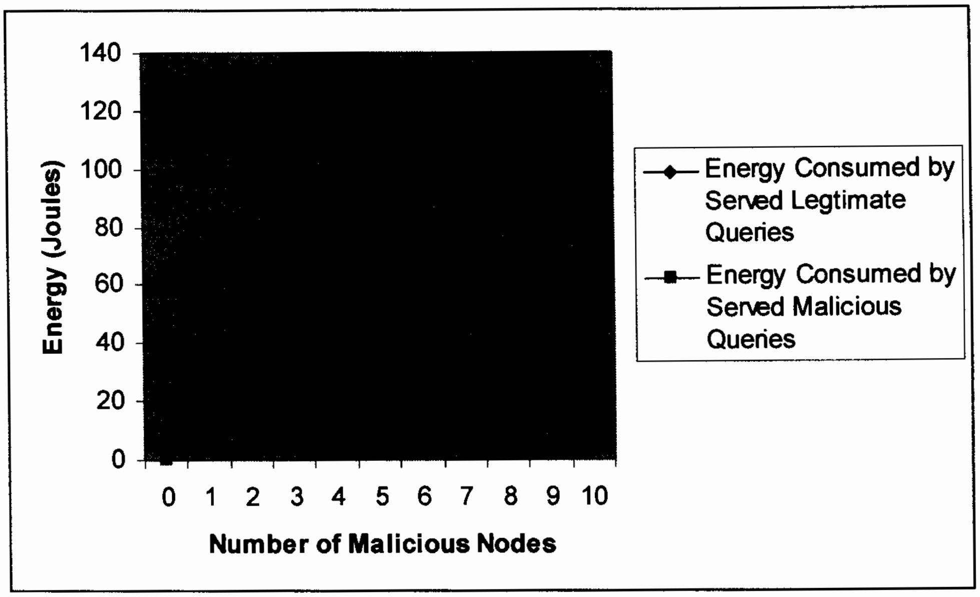

Figure 5.20: Energy Consumption by Served Queries-Fractional QDS

5.5.3 Energy Consumption in Weighted-Inverse QDS:

The energy calculations are shown in the figure 5.21. In case of zero malicious nodes energy consumed to serve legitimate queries was 96 joules. In the scenario when out of total 15 nodes 10 nodes were malicious the energy consumed by the served queries of legitimate nodes was 48 joules and energy consumed by the served queries of malicious nodes was 54 joules.

Pseudo Code for Getting Results of Graph 5.21:

```
1-We Conducted 11 experiments by changing the values of number of malicious nodes from 0 to 10.

2- For each experiment we:
    2.1- Calculate the initial amount of energy that the node has
    2.2- Calculate total queries sent by all the senders.
    2.3- Calculate that how many queries are served from each of the sender by using the Weighted Inverse query distribution technique (the value is calculated from the formula of this QDS).
    2.4- Queries are served according to the calculations of step 2.3.
    2.5- Calculate the current remaining energy.
    2.6- Check the status of each of the node (Malicious or Legitimate).
    2.7- Calculate the number of legitimate queries served and number of malicious queries served.
    2.8- Calculate the difference between values of step 2.1and 2.5.
    2.9- Divide the value of step 2.6 into two categories (Malicious and Legitimate) on the basis of number of queries served from legitimate node and malicious node.

3- We plot the graph between values of step 2.9 of all the 11 experiments.
```

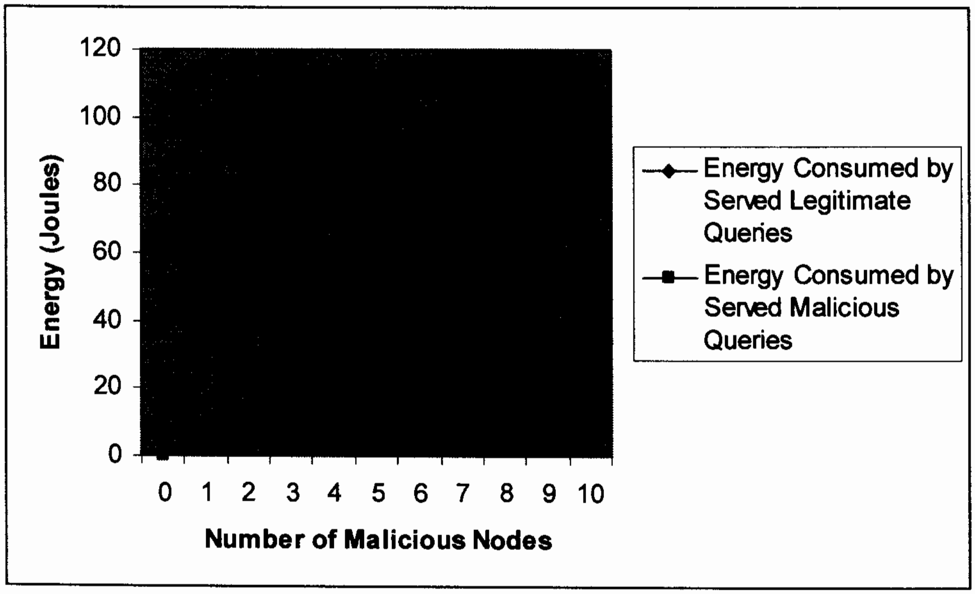

Figure 5.21: Energy Consumption by Served Queries-Weighted-Inverse QDS

5.5.4 Energy Consumption in Optimized Fractional QDS:

The energy calculations are shown in the figure 5.22. In case of zero malicious nodes energy consumed to serve legitimate queries was 132 joules. In the scenario when out of total 15 nodes 10 nodes were malicious the energy consumed by the served queries of legitimate nodes was 42 joules and energy consumed by the served queries of malicious nodes was 90 joules.

Pseudo Code for Getting Results of Graph 5.22:

```
1-We Conducted 11 experiments by changing the values of number of malicious nodes from 0 to 10

2- For each experiment we:
        2.1- Calculate the initial amount of energy that the node has
        2.2- Calculate total queries sent by all the senders.
        2.3- Calculate that how many queries are served from each of the sender by using the Optimized Fractional query distribution technique (the value is calculated from the formula of this QDS).
        2.4- Queries are served according to the calculations of step 2.3.
        2.5- Calculate the current remaining energy.
        2.6- Check the status of each of the node (Malicious or Legitimate).
        2.7- Calculate the number of legitimate queries served and number of malicious queries served
        2.8- Calculate the difference between values of step 2.1and 2.5
        2.9- Divide the value of step 2.6 into two categories (Malicious and Legitimate) on the basis of number of queries served from legitimate node and malicious node

3- We plot the graph between values of step 2.9 of all the 11 experiments.
```

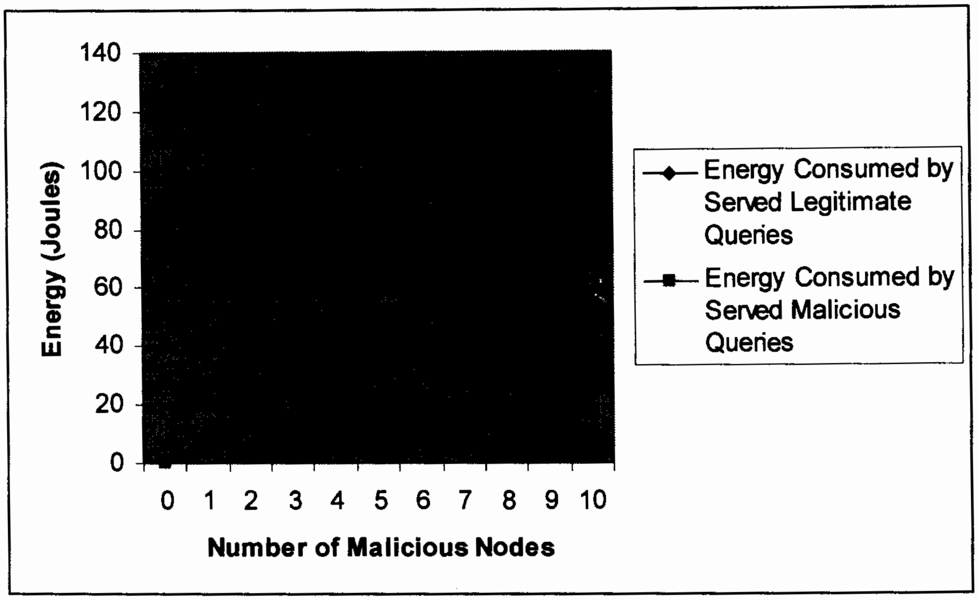

Figure 5.22: Energy Consumption by Served Queries-Optimized Fractional QDS

5.5.5 Energy Consumption in Double Trusted QDS:

The energy calculations are shown in the figure 5.23. In case of zero malicious nodes energy consumed to serve legitimate queries was 132 joules. In the scenario when out of total 15 nodes 10 nodes were malicious the energy consumed by the served queries of legitimate nodes was 54 joules and energy consumed by the served queries of malicious nodes was 78 joules.

Pseudo Code for Getting Results of Graph 5.23:

```
1-We Conducted 11 experiments by changing the values of number of malicious nodes
from 0 to 10.

2- For each experiment we:
    2.1- Calculate the initial amount of energy that the node has.
    2.2- Calculate total queries sent by all the senders.
    2.3- Calculate that how many queries are served from each of the sender by using
    the Double Trusted query distribution technique (the value is calculated from the
    formula of this QDS).
    2.4- Queries are served according to the calculations of step 2.3.
    2.5- Calculate the current remaining energy.
    2.6- Check the status of each of the node (Malicious or Legitimate).
```

2.7- Calculate the number of legitimate queries served and number of malicious queries served.
2.8- Calculate the difference between values of step 2.1and 2.5.
2.9- Divide the value of step 2.6 into two categories (Malicious and Legitimate) on the basis of number of queries served from legitimate node and malicious node.

3- We plot the graph between values of step 2.9 of all the 11 experiments.

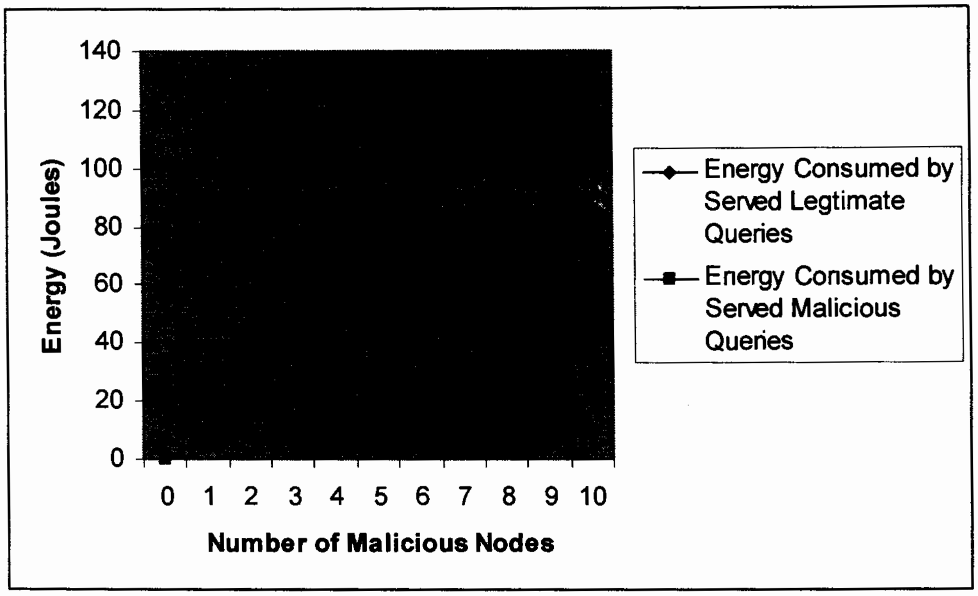

Figure 5.23: Energy Consumption by Served Queries-Double Trusted QDS

5.5.6 Energy Consumption in Complete Trusted QDS:

The energy calculations are shown in the figure 5.24. In case of zero malicious nodes energy consumed to serve legitimate queries was 132 joules. In the scenario when out of total 15 nodes 10 nodes were malicious the energy consumed by the served queries of legitimate nodes was 69 joules and energy consumed by the served queries of malicious nodes was 63 joules.

Pseudo Code for Getting Results of Graph 5.24:

1-We Conducted 11 experiments by changing the values of number of malicious nodes from 0 to 10.

2- For each experiment we:

2.1- Calculate the initial amount of energy that the node has.
2.2- Calculate total queries sent by all the senders.
2.3- Calculate that how many queries are served from each of the sender by using the Complete Trusted query distribution technique (the value is calculated from the formula of this QDS).
2.4- Queries are served according to the calculations of step 2.3.
2.5- Calculate the current remaining energy.
2.6- Check the status of each of the node (Malicious or Legitimate).
2.7- Calculate the number of legitimate queries served and number of malicious queries served.
2.8- Calculate the difference between values of step 2.1and 2.5.
2.9- Divide the value of step 2.6 into two categories (Malicious and Legitimate) on the basis of number of queries served from legitimate node and malicious node.

3- We plot the graph between values of step 2.9 of all the 11 experiments.

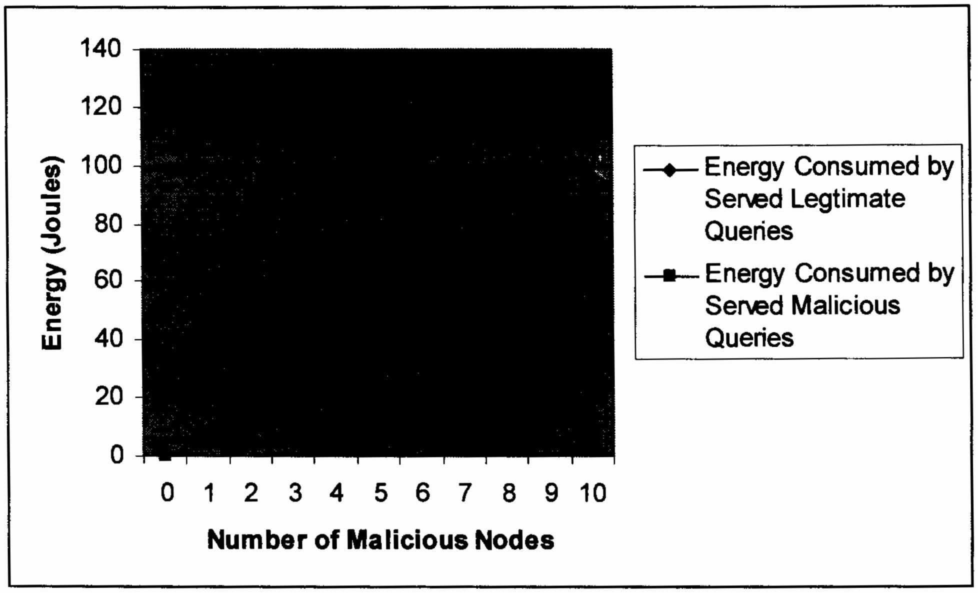

Figure 5.24: Energy Consumption by Served Queries-Complete Trusted QDS

5.5.7 Comparison of all QDSs in Terms of Energy Consumed by Served Queries of Legitimate Nodes:

In this experiment we compared all the six strategies in terms of energy consumed by served queries of legitimate nodes. We find that Complete Trusted QDS consumes maximum of its energy to serve the legitimate queries while weighted QDS consumes very low amount of energy to serve the legitimate queries. The initial energy consumption of weighted inverse QDS is different from the other five QDSs because in the other five QDSs the number of served queries are equal to the capacity of the receiver but in weighted inverse the number of served queries are less than the total capacity of the receiver. The results are shown in the figure 5.25.

Pseudo Code for Getting Results of Graph 5.25:

```
1- For each of the strategy
        2-We Conducted 11 experiments by changing the values of number of malicious
        nodes from 0 to 10.
        3- For each experiment we:
                2.1- Calculate the initial amount of energy that the node has.
                2.2- Calculate total queries sent by all the senders.
                2.3- Calculate that how many queries are served from each of the sender
                by using the specified query distribution technique (This value is
                calculated by the formula of specified QDS).
                2.4- Queries are served according to the calculations of step 2.3.
                2.5- Calculate the current remaining energy.
                2.6- Check the status of each of the node (Malicious or Legitimate).
                2.7- Calculate the number of legitimate queries served and number of
                malicious queries served.
                2.8- Calculate the difference between values of step 2.1and 2.5.
                2.9- Divide the value of step 2.6 into two categories (Malicious and
                Legitimate) on the basis of number of queries served from legitimate node
                and malicious node.
                2.10- Calculate the energy consumed by legitimate node's queries.

2- We plot the graph between values of step 2.10 of all the 11 experiments for all
strategies.
```

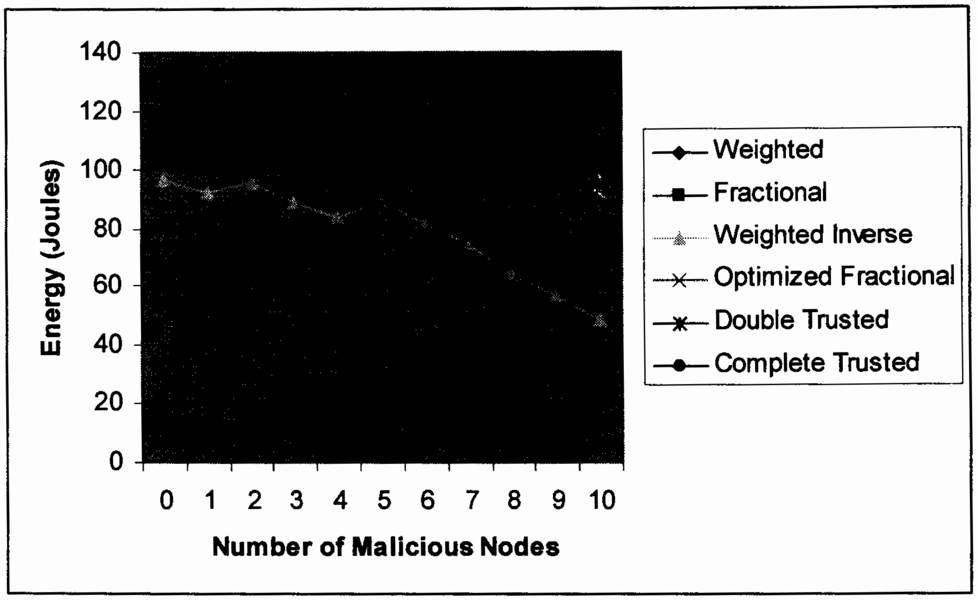

Figure 5.25: Comparison of all QDSs in Terms of Energy Consumed by Served Queries of Legitimate Nodes

5.5.8 Comparison of all QDSs in Terms of Energy Consumed by Served Queries of Malicious Nodes:

In this experiment we compared all the six strategies in terms of energy consumed by served queries of legitimate nodes. We find that Weighted Inverse QDS consumes very less amount of its energy to serve the malicious queries while weighted QDS consumes a large amount of energy to serve the malicious queries. Complete Trusted QDS is on the 2^{nd} number after weighted inverse which consumes very less amount of energy to serve the malicious queries. The results are summarized in figure 5.26.

Pseudo Code for Getting Results of Graph 5.26:

```
1- For each of the strategy
    2-We Conducted 11 experiments by changing the values of number of malicious
    nodes from 0 to 10.
    3- For each experiment we:
        2.1- Calculate the initial amount of energy that the node has.
        2.2- Calculate total queries sent by all the senders.
```

2.3- Calculate that how many queries are served from each of the sender by using the specified query distribution technique (This value is calculated by the formula of specified QDS).
2.4- Queries are served according to the calculations of step 2.3.
2.5- Calculate the current remaining energy.
2.6- Check the status of each of the node (Malicious or Legitimate).
2.7- Calculate the number of legitimate queries served and number of malicious queries served.
2.8- Calculate the difference between values of step 2.1and 2.5.
2.9- Divide the value of step 2.6 into two categories (Malicious and Legitimate) on the basis of number of queries served from legitimate node and malicious node.
2.10- Calculate the energy consumed by malicious node's queries.

2- We plot the graph between values of step 2.10 of all the 11 experiments for all strategies.

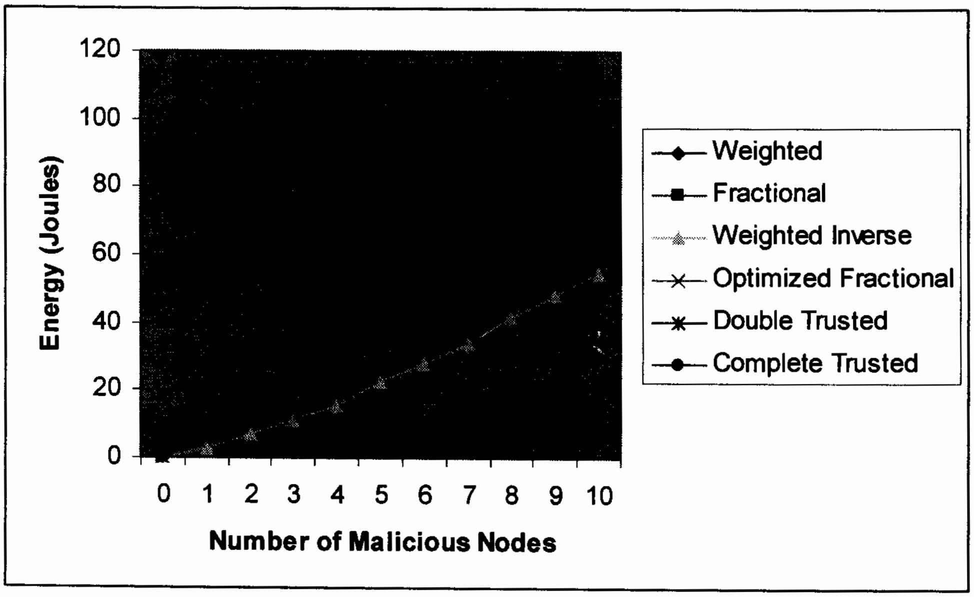

Figure 5.26: Comparison of all QDSs in Terms of Energy Consumed by Served Queries of Malicious Nodes

5.6 Average Delay:

The third performance parameter that we consider is the average delay in queries. Here first we calculate the average delay of all the queries of one QDS. After that we again take average of delays of all the strategies. So at the end we get only two values. One is showing the average delay of legitimate queries and second is showing average delay of malicious queries. Results are summarized in the figure 5.27. The maximum average delay for the legitimate queries is 1.8 seconds and in case of malicious nodes it is 4.3. The difference is due the strategy we adopted. Whenever serving the queries the server prefer the queries of the node that is sending minimum number of queries so it automatically reduce the average delay of legitimate queries and increase that of malicious queries.

Pseudo Code for Getting Results of Graph 5.27:

```
1- For each of the strategy
        1.1- We take the incoming queries and note their arrival time.
        1.2- Sort the serving queue in such an order that the node who sends minimum
        number of total queries is kept in front.
        1.3 Serve the queries accordingly.
        1.4- Note the serving time.
        1.5- Check the status of the node.
        1.6- Calculate the delay of legitimate queries and calculate their average.
        1.7- Calculate the delay of malicious queries and calculate their average.
2- Take average of all the strategies for the values of step 1.6.
3- Take average of all the strategies for the values of step 1.7.
4- Plot the graph between values of step 2 and 3.
```

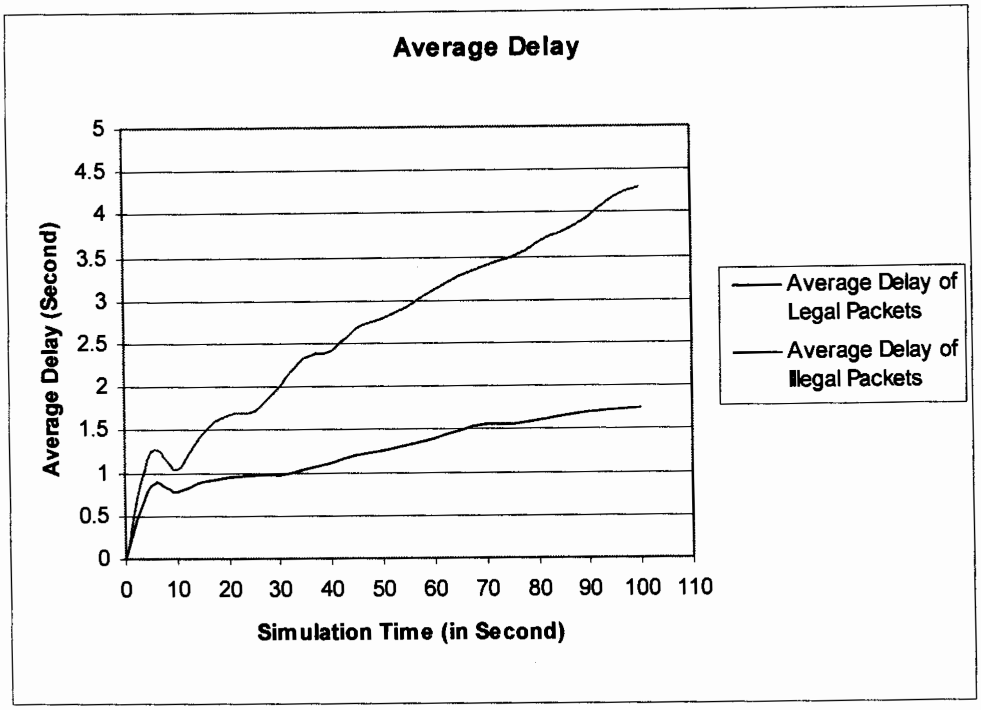

Figure 5.27: Average Delay

5.7 Chapter summary:

We have deeply studied the performance of all the Query Distribution Strategies in this chapter by implementing all of them in NS-2. There performance evaluation is shown here by different graphs constructed calculating three things here which are Remote Work, Energy and Delay. These are the parameters against which each of the strategy is tested and evaluated.

We have studied the performance of all these strategies regarding both Blind and Intelligent flooding and have observed the variations in there working regarding both aspects. With he increase in number of malicious node in the network the varying characteristics and working of all the strategies is closely examined to see which strategy is best under which situation or scenario. For some scenarios we see weighted QDS working efficiently and progressively and under some scenarios we come to see that Complete Trust QDS or Double Trust QDS performs more efficiently then any other strategy with consuming lesser resources. In the similar fashion all such scenarios are mentioned here to evaluate all these strategies for there performance visualizations.

6. Conclusion and Future Work

6. Conclusion and Future Enhancements:

We have finally concluded our dissertation here, giving the essence of our research. Together with the changes and improvements that can be made to this work to cope with the dynamic environment of this word especially in the field of networks.

6.1 Conclusion:

As we know from last few years among the wireless technology MANETs have gained much of the popularity. But this type of network is subjected to various kinds of attacks and threats; especially the most damaging are the Denial- of- Service attack. Securing such type of network is itself challenging due to the dynamic and infrastructure-less nature of MANETs. In our dissertation we have dealt with Application layer query flood DoS attack. This attack is quiet more damaging, where a single malicious query can contain the entire network and cause starvation and delay at the legitimate user's end.

We can use different preventive, detective and recovery techniques to tackle with such attacks but as we know these techniques might not be perfect and may fail to prevent the entry of any malicious node in to the network and its prevailing activities remain un-detective by the detection techniques. So in the mean while till this node can be disconnected from the network, we need such mechanism which can contain the amount of damage this malicious node can bring to the network. Hence we need such attack containment techniques that can minimize the effects during the attack time.

So in order to mitigate the effects of query flooding attacks during the attack time we have introduced the idea of Query Distribution techniques. These techniques also introduce fairness of resources among the nodes of the network. i.e. it allocates the resources on merit bases.

Fair distribution of resources minimizes the chances of starvation and discourages the attackers to occupy all the resources. These techniques are also used to balance the load of the network, by controlling the rate of query injection at each node.

We have formulated different query distribution techniques and have analyzed the performance of these strategies under the impact of single and multiple attackers using both types of flooding i.e. simple and intelligent flooding separately.

All these query distribution strategies are based on simple mathematical models which require simple processing capabilities, thus providing an equal amount of share to all of the active nodes of the network. By these strategies the malicious nodes also get their share of processing resources but the attacks effects are contained during the attack time and the network remains functional, continues to operate even in the light of the attack.

To achieve this target we have evaluated these different query distribution techniques in NS2 in order to find that which technique handles the flooding attacks in most appropriate way and under what circumstances. Of these techniques which one is suitable under what conditions. We have also evaluated that which technique consumes more energy and which consumed the minimum amount of energy.

6.2 Future Work:

In this thesis we have focused on containing the application layer DoS attacks i.e. Query flood attack. Enhancement in any area is always welcomed. Much of the work can be still done regarding the security of MANETs, and the area of attack containment can be more refined as some points are discussed as below.

In our work we have tested the Query Distribution Strategies against the isolated malicious nodes. These strategies can be tested in an environment where colluding malicious nodes exist, i.e. where malicious nodes work in a group by coordinating with each other in order to maximize their effects against the respective operating distribution strategy. We could then determine the performance of our these strategies under colluded malicious nodes impact and can also analyze this that whether all of these strategies could efficiently cater with this colluded attack or need some modifications to cope with this situation.

Much of the work is already done in the area of attack detection. Where as how the detection techniques and the containments techniques can both work in collaboration and can be benefitted from each other. This is an open area for future considerations.

Vehicular Ad hoc Networks is one type of MANETs .These strategies can be enhanced to contribute to containment issues in Vehicular Ad hoc Networks.

References

[1] William Stallings, Cryptography and Network Security principles And Practices, Fourth Edition.

[2] Wikipedia: The biggest Encyclopedia. www.en.wikipedia.com.

[3] Chakeres, et al. Internet-Draft on MANET Architecture October 2006. RFC 3753, June 2004. RFC 2461, December 1998. RFC 2501, January 1999.

[4] Neil-Daswani "Denial-of-Service (DOS) Attacks and Commerce Infrastructure In Peer-to-Peer Networks", 2005

[5] N. Daswani, H. Garcia-Molina, "Query-Flood DoS Attacks in Gnutella", ACM Conference on Computer and Communications Security, Washington, DC, November 2002.

[6] Neil Daswani and Hector Garcia-Molina, "Blasting in Chord", Stanford CS Technical Report, January 2005.

[7] N. Daswani, H. Garcia-Molina, "Pong-Cache Poisoning in GUESS", ACM Conference on Computer and Communications Security, Washington, DC, October 2004.

[8] N. Daswani, H. Garcia-Molina, B. Yang., "Open Problems in Data-Sharing Peer-to-Peer Systems" International Conference on Database Theory, Siena, Italy, January 2003.

[9] Mayank Bawa, Brian F. Cooper, Arturo Crespo, Neil Daswani, Prasanna Ganesan, Hector Garcia-Molina, Sepandar Kamvar, Sergio Marti, Mario Schlosser, Qi Sun, Patrick Vinograd, Beverly Yang, "Peer to Peer Research at Stanford", ACM SIGMOD Record. http://www-db.stanford.edu/~bawa/./Pub/stanford.ps

[10] Qixiang Sun, Neil Daswani, Hector Garcia-Molina, "Maximizing Remote Work in Flooding-based Peer-to-Peer Systems" Journal of Computer Networks, Elsevier, Science Direct Publishers, 2006.

[11] Q. Sun, N. Daswani, H. Garcia-Molina, "Maximizing Remote Work in Flooding-based Peer-to-Peer Systems", 17th International Symposium on Distributed Computing (DISC 2003), Sorrento, Italy, October 2003.

[12] Sunsook Jung, Nisar Hundewale, Alex Zelikovsky, "Energy Efficiency of Load Balancing in MANET Routing Protocols", IEEE, 2005.

[13] H. Wu et al., "Mddv: A Mobility-Centric Data Dissemination Algorithm for Vehicular Networks," Proc.1st Int'l. Wksp. Vehic. Ad Hoc Networks (VANET), Oct. 2004, pp.47–56.

[14] Min Shen , Dongmei Zhao, ”Throughput Analysis of IEEE 802.11 and IEEE 802.11e MAC”,ACM, The Third International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks, August 7-9 2006, Waterloo, ON, Canada.

[15] S-Y. Ni, Y-C Tseng, Y-S Chen and J-P Sheu, “The broadcast storm problem in mobile ad hoc networks”, Proc. ACM MobiCom ’99, Seatle, USA, Aug. 1999.

[16] B. Williams and T. Camp,”Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks”, ACM MOBIHOC, 2002.

[17] Joon Yoo, Hong-ryeol Gil, Chong-kwon Kim,” INK: Implicit Neighbor Knowledge Routing in Ad Hoc Networks”, Vehicular Technology Conference, IEEE 2003.

[18] Gokhan Korkmaz, Eylem Ekici, “Urban MultiHop Broadcast Protocol for InterVehicle Communication Systems”, ACM VANET’04, October 1, 2004, Philadelphia, Pennsylvania, USA.

[19] Dragos¸ Niculescu and Badri Nath, “Trajectory Based Forwarding and Its Applications”, ACM MobiCom’03, September 14–19, 2003, San Diego, California, USA.

[20] Timo Kosch, Christian J. Adler, Stephan Eichler, Christoph Schroth, and Markus Strassberger ,” the scalability problem of vehicular ad hoc networks and how to solve it”, IEEE Wireless Communications , October 2006.

[21] Tatsuaki Osafune, Lan Lin, Massimiliano Lenardi,” Multi-Hop Vehicular Broadcast (MHVB)”, 6th International Conferences on ITS Telecommunications, 2006.

[22] Chi-Kin Chau, Jon Crowcroft, Kang-Won Lee, Starsky H.Y. Wong ” Inter-Domain Routing for Mobile Ad Hoc Networks” , 2008.

[23] Sasu TarkomaDirk, TrossenMikko ,Särelä, “Black Boxes: Making Ends Meet in Data Driven Networking” , 2008.

[24] Björn Scheuermann,_ Matthias Transier, Christian Lochert, Martin Mauve , Wolfgang Effelsberg, ” Backpressure Multicast Congestion Control in Mobile Ad-Hoc Networks”, 2008.

[25] Rüdiger Schollmeier, Ingo Gruber and Michael Finkenzeller, “Routing in Mobile Ad Hoc and Peer-to-Peer Networks A Comparison”, 2002.

[26] Robert Castaneda, Samir Das and Mahesh K Marina, “Query Localization Techniques for On-Demand Routing Protocols in Ad Hoc Networks”, 2004.

[27] Mohamed Abdelhafez, "Modeling and Simulations of Worms and Mitigating Techniques", 2007.

[28] Bo-Chao Cheng, Huan Chen and Ryh-Yuh Tseng, "A Good IDS Response Protocol of MANET Containment Strategies" IEICE Transactions on Communications, 2008.

[29] Aniruddha Chandra , "Ontology for MANET Security Threats"
Electronics and Telecommunication Engineering Department Jadavpur University
Kolkata 700032, India.

[30] Antonio Martin, Jeffrey Smith and Manfred Koethe , "A Platform Independent Model and threat Analysis for Mobile Ad hoc Networks", 2007.

[31] Antonio Martin, "A Platform Independent Risk Analysis for Mobile Ad hoc networks"
Boston University Conference on Information Assurance and Cyber Security, Dec 2006.

[32] Radha Poovendran, "Information Assurance in MANET and Sensor Networks" University of Washington. 2005.

[33] Yihong Zhou, dapeng Wu and Scott M. Nettles "On MAC-layer denial of service attacks in IEEE 802.11 ad hoc networks: analysis and counter measures" International Journal of Wireless and Mobile Computing, 2006.

[34] Sudipto Das "Security Issues in Mobile Ad hoc networks, Department of science and Engineering Jadavpur University, 2006.