COUNTER ATTACK AS A DEFENSE MECHANISM

IN AD HOC MOBILE WIRELESS NETWORKS

By

BIPUL CHANDRA

Master of Science in Computer Science

Oklahoma State University

Stillwater, Oklahoma

2011

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
July, 2011

COUNTER ATTACK AS A DEFENSE MECHANISM
IN AD HOC MOBILE WIRELESS NETWORKS

Thesis Approved:

Dr. Johnson Thomas

ThesisAdviser

Dr. Subhash Kak

Dr. Michel Toulouse

Dr. Mark E. Payton

Dean of the GraduateCollege

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

CHAPTER I

INTRODUCTION

1.1 **Motivation**

The merit of having an infrastructure less network was first discovered in the 1970s. At that time, Computers were bulky and so were the radio transreceivers. Since then, the technology of both computers and radio communication has improved many folds. This exceptional growth gave birth to the wireless network. With rapid development in wireless communication technology, Ad-Hoc Mobile Wireless Network (MANETs) have emerged and evolved in many forms [12]. MANETs are rapidly gaining popularity because they do not rely on a pre-infrastructure and can be deployed spontaneously. Application of MANETs ranges from offices to modern battlefields. The distributed nature of MANETs has eliminated the need for centralized authentication and monitoring.

However, compared to wired networks, MANETs are more vulnerable to security attacks due to their unique features, such as stringent power consumption, error prone communication media and highly dynamic network topology [12]. Confidentiality, integrity and availability are three major requirements for any information security systems. To achieve confidentiality and integrity, cryptography solutions of wired networks can be used with little or no change. However, the security of MANETs has been challenged by covert manipulation of communicating network entities due to usage of lightweight protocols of MANETs. Denial of Service (DoS) attacks can also be successfully launched due to the lightweight protocols and energy restrictions of MANETs. The limitations of MANETs resources prevent the implementation of

sophisticated security measures in a MANET. Securing a MANET therefore is a challenging task. Although, a number of papers have been focused on securing and guarding against attacks, as far as we are aware, nobody has investigated counter attacks as a defense mechanism in MANETs.

1.1.1 Problems with existing approaches

Since security is very important in MANETs, numerous investigations have been done on security issues in MANETs. Some have focused on key management; others have focused on identifying a specific type of attack and measures to stop it. Intrusion prevention mechanism like encryption of message and authentication can be used in MANETs to reduce intrusions, but cannot eliminate them completely. The history of security has mostly focused on developing defensive mechanisms, such as Firewalls, Gateways etc, but very little work has looked at offensive measures to be taken post detection. Currently existing offensive responses are developed to handle a specific type of attack. But there does not exist a general offensive mechanism that can be used in any attack situation. Rather than replacing traditional defensive mechanisms, offensive counter mechanisms will complement these approaches thereby strengthening the security of the MANET.

1.1.2 Proposed Approach

In this thesis, we propose three counter attack models, namely, Round Robin attack, Self-Whisper attack and flooding attack. The goal of all these attacks is to use up intruder critical resources like energy, communication, processing, storage and thereby force the intruder to eventually enter into a DoS status.

The counterattack models will depend on the goals of the offensive response. If the goal is to learn about what kind of information the intruder is looking for, different attack models can be used. The goal will be realized via agent nodes. An agent node is a dedicated, specialized node whose job is to carry out the DoS or other attack against the intruder in coordination with other agent nodes deployed and distributed across the MANET. The agent nodes can communicate with each other using multicast.

Some important advantages an offensive approach has against an intruder are:

I. **Psychological Advantage**: Once an adversary node knows that there are nodes inside the target network which can retaliate, this gives the attacker an impression that the target network is well protected, forcing it to think twice before attacking the network or continuing with the attack. Such a situation may force the adversary node to either retreat or make a mistake which can be further exploited by agent nodes. Hence, our proposed approach can drastically reduce the chances of attack in the first place and improve network security.

II. **Added Layer of Security:** An offensive approach adds an extra layer of security to the defensive measures in place.

III. **Reduced Rate of Successful Attack**: Another advantage of the proposed approach is that it significantly reduces the rate of successful attacks against the network. In order to attack the network, an adversary node first needs to defeat the agent nodes, then penetrate the defensive wall of Firewalls and Gateways. This makes an attacker's task more complicated and risky.

IV. **Learn about Attacker Resources:** Another advantage of going offensive is that, it helps to extract knowledge about the attacker such as the power of the attacker in terms of critical resources like bandwidth, processing, storage capacity, and power. Knowing about these critical resources help the agent nodes to formulate an effective strategy against the adversary node.

V. **Buy Time:** Going offensive against the attacker will slow down the attack and hence will buy more time to organize agent nodes, strengthen the security of the network and formulate an effective strategy against the attacker.

VI. **Waste Attacker Resource:** One important advantage of an offensive approach against an attacker is that, it wastes **critical resources of the attacker, like power, storage, bandwidth and processing**. Most of the devices that operate in an ad hoc mobile network are **battery operated**. These devices have limited power supply and hence the more work they have to do in terms of processing or transmitting data, the sooner they will consume their power.

1.2 Research Objectives

The overall objective of this thesis is to study the feasibility and effectiveness of counter attacks in a MANET. The effectiveness of these models will be measured based on the following three parameters:

- Time taken to neutralize/marginalize the intruder node.
- Energy consumed to neutralize/marginalize the intruder node.
- Damage done to intruder in terms of resources.

The research objectives are as follows:

- Review the general security requirements of MANETs.
- Identify and classify the major attacks in MANETs.
- Explore the existing counter measures against classified attacks.
- Propose and study counter attack models that can be used against any kind of intruder attack effectively.
- Develop a framework of counter attack models.
- Design algorithms to carry out the counter attack for each proposed model.
- Simulate the proposed models and compare them to find out the best possible models, if any.

1.3 Research Contribution

We study the feasibility and effectiveness of our proposed counter attack model for the security of MANETs. DoS attack is used as the primary tool for an attacking intruder node. The counter attack will be carried by surrounding the intruder node with agent nodes where each agent node will first position itself into direct radio transmission range of the intruder node before launching a DoS attack. This will ensure that the effect of counter attacks should not disrupt the normal functioning of the network.

1.4 Outline of Thesis

The remainder of this thesis is organized as follows: Chapter 2 gives an overview of MANETs and their potential applications. In addition, it describes the security issues,

types of attack and countermeasures in MANETs. Chapter 3 proposes our counter attack model. Chapter 4 evaluates the counter attack model with simulations. Finally, a summary of this thesis and a discussion of future work are presented in Chapter 5.

CHAPTER II

REVIEW OF LITERATURE

2.1 Ad Hoc Mobile Wireless Network

A mobile ad-hoc wireless network (MANET) is a collection of two or more wireless devices equipped with wireless communications capability that does not have any fixed infrastructure or centralized authentication system. Such devices can communicate with another device that is immediately within their radio range or outside their radio range through relay nodes [12]. A mobile ad-hoc wireless network is *self organizing* and adaptive. Since MANETs do not rely on any network entities, MANET can be formed or de-formed on the fly without any additional infrastructure. Since any device equipped with wireless communication can join the ad-hoc network, there is a vast heterogeneity among devices. This vast heterogeneity among devices means that communication, storage, computation and power consumption of these devices also vary tremendously [12].

To facilitate the communication in ad-hoc wireless network, many protocols have been developed. But none of these protocols have yet been standardized. However, one protocol in particular is gaining popularity and maybe standardized soon. The Ad-hoc On-Demand Distance Vector (AODV) routing protocol allows on-fly formation of network. It allows users to find and maintain routes to other users in the network when such routes are needed. The AODV routing protocol provides unicast, multicast and broadcast communication in ad-hoc mobile networks [4].

Figure 2.1 Heterogeneous mobile device ad hoc networks

2.2 Challenges Facing Ad Hoc Mobile Wireless Network

Ad hoc networks face many major challenges. We will focus on challenges related to Routing, Energy Efficiency, and protocols. Some of these major challenges are 12]:

1) **Media Access**

   Given the fact that MANETs lack centralized control and there is no static node, the MAC protocol must contend for access to the channel while at same time avoiding possible collision with neighboring nodes. In addition, the problem of hidden and exposed terminals must be accounted for when designing the MAC protocol for MANETs.

2) **Routing**

   The typical distance vector routing protocols of wired network cannot work in the highly dynamic and in deterministic topology of MANETs. The typical multicast protocols in wired network will not work with MANETs. Multicast protocols of wired network work because nodes are static in nature, unlike MANETs.

3) **Energy Efficiency**

   Forwarding packets on behalf of others will consume power, and this can be quite significant for nodes in an ad hoc network. Hence, power unaware protocols of wired network are not effective for an ad hoc network. Moreover, the battery technology is still lagging behind microprocessor technology. A typical LI-Ion battery will last 2-3 hrs. Hence, energy conservation is the most important factor for a node in an ad hoc network.

4) **TCP performance**

TCP relies on measuring round trip time (RTT) and packet loss to conclude if congestion has occurred in the network. Unfortunately, TCP is unable to differentiate between node mobility and network congestion. Mobility of node in an ad hoc network may result in either packet loss or longer RTT. Hence, some enhancement is required for TCP to work well in an ad hoc network.

5) **Service Location, Provision and Access**

The ad hoc network consists of heterogeneous devices and not all of them are capable of playing the role of server. This means that traditional client/server RPC will not work in ad hoc networks.

6) **Security and Privacy**

As any node in an ad hoc network can play the role of router, it is important to make sure that the node is authentic and only authentic nodes are forwarding packets. It is very easy for a node to deceive in an ad hoc network. It can manipulate the protocol by replaying false information such as the shortest distance to destination. Attacks in MANETs include Sinkhole/Black hole, Sybil, wormhole attacks.

2.3 Applications

1. **Office**

Mobile ad hoc devices can automatically recognize the presence of other devices through sensing the presence of neighboring beacons. This will allows the synchronization of devices and transfer of emails, files, personal calendar seamlessly from handheld devices to desktop.

Figure 2.2 Mobile Devices talking to each other.

2. **Traveling**

   A passenger carrying a personal wireless ad hoc device which on entry to the airport terminal will be able automatically communicate with the air line system via an ad hoc wireless access point. Since the ticket is already booked, an electronic version of the boarding pass or confirmation seat number can then be assigned and conveyed to the passenger, thus. eliminating the need to stand in a queue.

3. **Home**

   A user ad hoc device can communicate with home wireless device to perform various tasks on behalf of the user. For example, ad hoc devices worn by different family members can be programmed to have different levels of control and setting for house hold electronic devices.

Figure 2.3 Smart Home

4. **Shopping Mall**

   The shopping malls of the future will have their products installed with cheap RF tags. A customer carrying a handheld wireless device can read the product price and related information even without entering the shop.

5. **Modern Battlefield**

   Ad hoc networks are known as self organizing networks. Through multi hop communications, soldiers can communicate to remote soldiers via data hopping or data forwarding from one radio device to another. Through ad hoc networks, it is possible to wirelessly manage the minute sensor networks which are scattered throughout enemy territory.

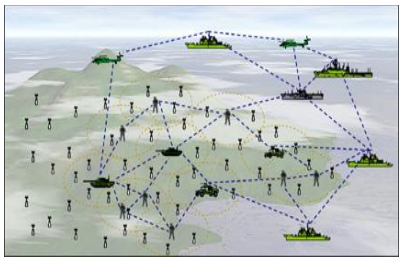


Figure 2.4 Battlefields

6. **Location/Context based Service**

   When a user equipped with ad hoc communication device enters a shopping mall, the shopping mall ad hoc network will automatically send him fast selling products, information about dining, movies currently running in theaters etc. Analogously, the same can be done when a user visits a museum, airport etc.

2.4 Limitations

Ad hoc nodes are typically characterized by limited power supplies, small memory size and limited computational ability. Ad Hoc wireless networks typically use a low bandwidth because communication bandwidth is very expensive, consuming large amounts of energy and processing power. Unreliable communication is another threat to Ad Hoc network security. Packet-based routing of the Ad Hoc network is connectionless unreliable transfer. In high density MANETs, because of the Broadcast nature of MANETs, nodes can interfere with each other's communications. The large amount of

latency in the network is caused by multi-hop routing, network congestion, and node processing [4]. Hence, communication between nodes in a MANETs is unreliable.

2.5 Security Requirement of MANETs

Nodes in Ad Hoc networks need to have the following capabilities to successfully deal with attacks.

- **Light weight Encryption Algorithms**: They need an encryption algorithm that consumes little power but gives high standard data encryption.
- **Time Synchronization**: Nodes in ad hoc network can periodically synchronize their timer. This will help to track any attack in the network by calculating the round trip timer (RTT).

Because of the limitations of ad hoc networks, supporting secure communication in such a network is a great challenge. The general security requirements of ad hoc networks are as follows:

- **Data Confidentiality**: The nodes communicating to each other may want maintain their data privacy from neighbor nodes.
- **Data Authentication**: Nodes needs to make sure that data that they receive originate from an authentic source, not from an adversary
- **Data Integrity**: Ad hoc networks need to make sure that data has not been altered by an adversary node.
- **Data Freshness**: Ad hoc network needs to make sure that messages are not retransmitted or replayed.
- **Robustness**: Ad hoc networks have to be robust in nature to minimize the impact of any successful attack.

2.6 Types of Attacks and Counter Measures

2.6.1 Wormhole Attack

In the wormhole attack, a malicious node picks the packet from one location of the network and tunnels it to another malicious node at another location in the network,

which replays it locally [1]. The wormhole tunnel can be established in many ways such as packet encapsulation, out of band channel, high power transmission, Packet Relay, Protocol Deviation [1]. The Wormhole attack exploits one of the features of the ADOV protocol where the AODV protocol allows a node to choose the shortest route from source to destination, when multiple routes have been discovered during the "route discovering "phase of the protocol [4]. To counter the wormhole attack Yih-Chun, Adrian and David [5] proposed packet leashes, where a leash is any information that can be added to the packet designed to limit the maximum allowed transmission distance [5]. There are two categories of packet leashes: 1. Geographical Leash 2. Temporal Leash. A geographical leash makes sure that the recipient is at a certain distance from sender. On other hand, a temporal leash puts an upper bound on how far the packet can go. A Geographical leash can be created if each node knows its own location as well as some rough estimation about the receiver location. All the nodes need to have some kind of loosely synchronized clock which allow them to validate each other packet timestamp. On the other hand, a temporal leash requires that each node in the network must have a tightly synchronized clock with the maximum difference in each other's clock not exceeding few micro or nano seconds [5]. However, in certain circumstances, even packet leashes will not prevent a wormhole attack, such as when the sender and receiver are not within transmission range of each other [5].

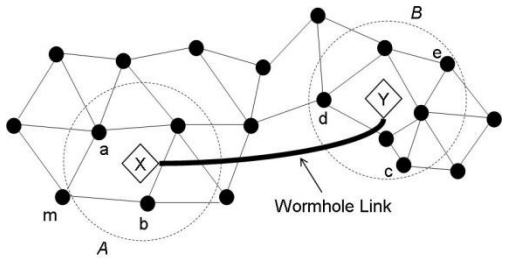


Figure 2.5 Wormhole Attack

2.6.2 Black hole/Sinkhole Attack

The original AODV protocol operates on the assumption that all the nodes in the network are trustworthy [4]. AODV allows a sender to always choose the shortest path to the destination [4]. This underlying assumption about the trust environment of the network

can easily be exploited by a malicious node. Because the malicious node does not have to check its routing table before replying to RREQ, the reply from a malicious node will always be faster than other nodes. On receiving the reply from a malicious node, the node which started a route discovery process, will conclude that route discovery is over and hence it will establish a route which actually runs through the malicious node. Any traffic through the malicious node can be lost or compromised. A Black hole attack is a kind of denial of service attack where a malicious node attracts all traffic falsely and drops them without forwarding them to the destination [3].

To counteract this attack, Martin at el in [6] presented the idea of a *watchdog.* The watchdog method detects misbehaving nodes if a node in the route fails to forward the send packet within a time ***t*** set by sender. However, the authors fail to note that if a node is heavily loaded, it is possible that the time taken to forward the packet may be beyond the timeout period ***t*** to forward the packet. In this case, as per ***t*** [6], the node will flag as "misbehaving" which is not necessarily true. Also, a malicious node may overload an ordinary node intentionally so that it flagged as a "bad" node or a group of malicious nodes intentionally overload a part of the network and then partition the network based on the claim that most of the nodes in that portion of the network are flagged as "bad". However, Animesh et al. [7], extended the idea of Martin [6], where they classify the nodes in the network into three categories, watchdog, ordinary and trusted. Few initial nodes that join the network are trusted nodes or in other words "good" nodes. Watchdog nodes must be selected from the group of trusted nodes only. Any node that joins the network thereafter will join as an ordinary node. The watchdog node continuously keeps tracking *SUSPECT_THRESHOLD* and *ACCEPTANCE_THRESHOLD* thresholds. If an ordinary node crosses the set *SUSPECT_THRESHOLD*, it declared as a "malicious" node. On the other hand, if a node crosses its *ACCEPTANCE_THRESHOLD*, it will be promoted to the group of trusted nodes. Also, watchdog nodes are selected for specific period of time only. Once the time period expires, a new set of watchdog nodes need to be selected from the group of trusted nodes.

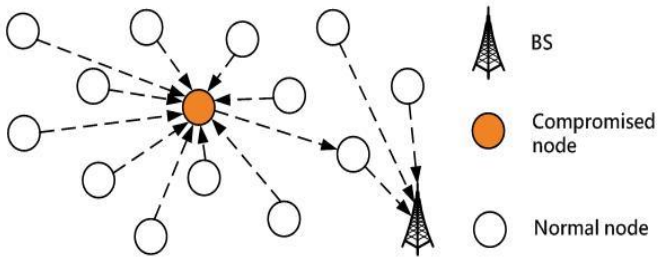


Figure 2.6 Black hole/Sinkhole Attack

2.6.3 Sybil Attack

The Sybil attack was first reported by Douceur in the context of peer-to-peer networks [8]. In the *Sybil* attack, a node illegally claims multiple identities. In the worst case, an attacker may generate any number of nodes by using just one physical device. The Sybil attack can be launched in various forms, direct vs. indirect communication, fabricated vs. stolen identities and simultaneous attacks. There are several known Sybil attacks such as *Distributed attacks, Routing attacks, Data Aggregation, Voting, Fair resource allocation, Misbehavior detection* [8]. Karlof and Wagner [9] pointed out that the Sybil attack can be used against *multipath* routing protocols in MANET. It is possible that all disjointed path might actually be going through a single malicious node. All the above mentioned attacks can be launched because the Sybil node simply overcame the ordinary nodes.

To counter the attack Douceur [8] proposes resource testing as a method of direct verification of node identity. In resource testing, it is assumed that each node is limited in physical resources. The resource proposed by Douceur for this purpose is computation, communication and storage. But any of these resources can be manipulated by a malicious node, because, a malicious node is assumed to generally have large amounts of storage, computation and communication capabilities. However, James, Elaine, Dawn and Adrian in [2], proposed some new defenses against Sybil attack like Radio resource testing, verification of key for random key redistribution, registration and position verification.

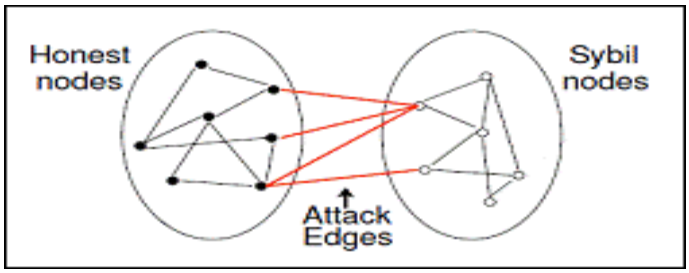


Figure 2.7 Sybil Attack

2.6.4 Denial of Service (DoS) Attack

The "Ad Hoc Flooding Attack (AHFA)" [10] can result in a DOS attack if used against reactive routing protocols such as DSR and AODV. The AODV protocol has some sort of inbuilt defense against a DOS attack. A node cannot originate more than *RREQ_RATELIMIT* messages per second. Also, after originating a RREQ packet, a node will wait for a route reply (RREP) packet which has a *TTL* (Time to live) associated with it. A node can re-broadcast RREQ packet only if it will not receive RREP packet within a specified TTL time. However, an adversary node may violate all the rules and broadcast mass route requests (RREQ) with higher TTL values for a distant or non-existing IP addresses in the network. If the IP address does not exist in network, then RREQ packet remains in the network for a longer period of time. Consequently, all the routes in the network will be flooded with the adversary node's RREQ packet. The adversary's objective behind this mass broadcast is to exhaust the communication bandwidth of the network, hence depriving legitimate nodes from valid network communications. The ad hoc flooding attack in mobile wireless network is similar to SYN attack in wired networks.

To counteract this attack, Ping Yi [10] and his team have developed a defense against ad hoc flooding called *Neighbor Suppression.* In neighbor suppression each node computes the rate of RREQ. If a node's neighbor finds that the node RREQ rate has crossed a defined threshold, Rate_RREQ, it will blacklist the node and does not accept any RREQ packets originating from the node. Hence, if an Intruder tries to flood the network with mass broadcasting of RREQ packets, its neighbor will eventually Blacklist it as soon as the intruder node crosses the defined threshold.

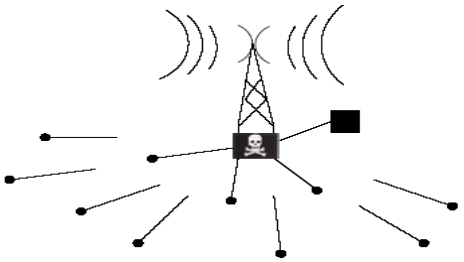


Figure 2.8 HELLO packets Flooding Attack

2.7 Comparative study of Simulators

There are many simulators for wireless sensor networks. Evaluating the strengths and weaknesses and choosing the proper simulator to realize the model is very important. Five simulators were selected and are compared below:

Aspect	J-Sim	OMNet++	NS-2	ShoX	NetLogo
Visualization	Nam Trace File No Own Tool	online with model inspection, to go back, simulation must be repeated	Trace file, can be viewed with nam	trace file, internal viewer	Dynamic plot, the command center can show the internal result
Statistic	online plot, exporting to file must be done by user	trace file, can be displayed with plove	log file, can be displayed with xgraph	statistics file, internal viewer or export to gnuplot	Dynamic plot, exporting to file
Strengths	flexibility Java based	maturity model inspection GUI support	model base user base	GUI support visualization architecture	GUI support visualization
Weakness	GUI support visualization capabilities	energy model MAC competitors	OTcl architecture	documentation lack of models	architecture

Table 2.1 Comparative Study of Simulators

NS-2 is the software which best meets our requirements. Our requirements are:

- A simulator with support for Mobile Networking and AODV ad-hoc protocol
- A simulator with visualization capability
- A flexible simulator that supports the development of new customized protocols
- A simulator with tools for recording and analyzing simulation

NS-2 supports the above requirements and was therefore chosen.

CHAPTER III

COUNTER ATTACK MODELS

This chapter focuses on the design issues of the proposed counter attack model and their operation in the real world. The main goal behind these counter attack models is to exhaust the intruder's resources like energy, communication bandwidth and hence force it either to leave the network or to ultimately die. However, the potential use of these models is not limited as an attack tool, but they can also be used to extract the intruder's interest, which will help network administrators to learn about intruder behavior.

3.1 Problem Specification

The counter-attack model is primarily focused on intruder presence and is independent of the attack launched by the intruder. The following assumptions have been made to build the counter attack models:

1. The System has already identified the intruder node inside the network.
2. Agent nodes are equipped to track any node inside the network in real time.
3. Agent nodes are part of a single group and use the multicast feature of AODV to communicate with each other. No other node can join the group.
4. Agent nodes always launch an attack in coordination with each other.
5. Before launching a counter-attack, agent nodes must position themselves in direct communication range of the intruder node.

6. All the nodes in the network including intruder are identical in terms of resource capabilities such as energy, communication range, communication bandwidth, processing & storage. All the nodes in the network have equal initial energy, equal communication bandwidth and equal radio transmission range.

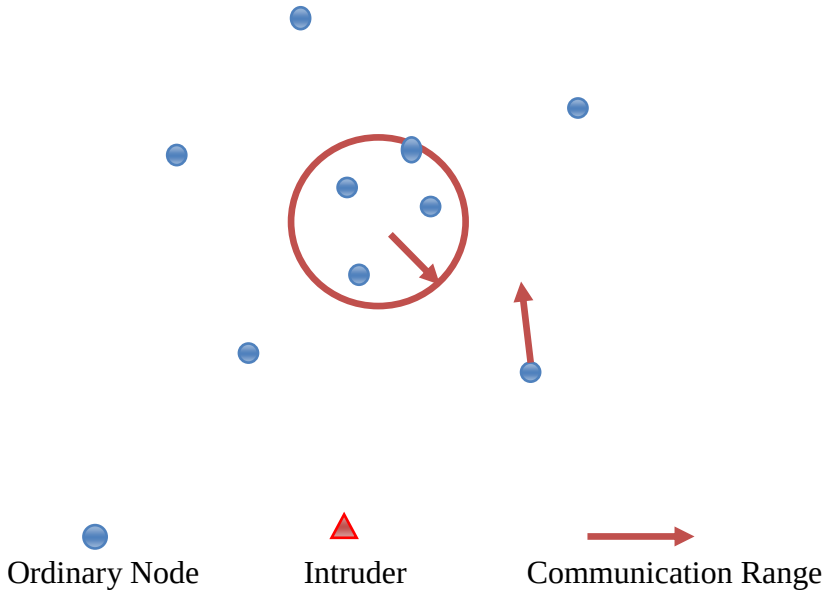


Fig 3.1 Network Model

3.3 Outline of Approach

This thesis primarily focuses on studying the effectiveness of proposed counter attack models against an intruder. The objectives of the different models are:

i. Minimize Time taken by a group of agent nodes to neutralize/marginalize the intruder node.
ii. Maximize energy consumption rate of the intruder node.
iii. Maximize the packet drop rate of the intruder node.

We will measure the time, average energy consumption and packet drop rates of agent nodes as well as those of the intruder.

Agent Node: These are dedicated mobile nodes with following features:

- equipped with capability to track any node inside network in real time

- part of an exclusive counter attack group
- passive in nature unless until not carrying out counter attack operations

Counter Attack Group: A group of agent nodes.

Ordinary Node: A node which is neither an agent node nor an intruder node.

Intruder Node: A node inside the network which is carrying an unauthorized activity

Each of three counter-attack modes is used to measure the three parameters mentioned above in this section. The models are:

I. **Round Robin Counter-Attack**: We will evaluate round robin on the three parameters identified above. The primary objective of this counterattack model is to make the attacker consume his energy, that is, objective two above. In particular we measure the average energy consumption rate of a group of agent nodes as well as that of intruder. Round robin allows only one of the agent nodes to carry a counter-attack at any given time, and as all the nodes are identical, the probability of an agent node to successfully launch a DoS attack against the intruder is very low. Hence the packet drop rate by the attacker is expected to be very low. Hence energy consumption is the main goal of this counterattack.

II. **Flooding Counter-Attack**: In this model, the prime objective is to quickly marginalize/neutralize intruder node (objective 1 above) and also to increase packet drop rate by intruder (objective 3), thereby quickly imposing a DoS on the attacker. Because flooding counter-attack allows multiple simultaneous communication channel to be opened through the intruder, all agents will start communicating with the intruder at the same time. This sudden rise in traffic should consume intruder energy and communication bandwidth faster than the round robin. Hence, time taken in this model to neutralize intruder should be less than the round robin counter-attack model.

III. **Self-Whisper Counter-Attack:** Because self whisper is a hybrid of the round robin and flooding counter-attack models, where multiple pairs of communication channels can be opened through the intruder and each pair of communication will

occur in round robin fashion, the objective of this attack is to achieve all three goals as identified above.

The above three counter attack models will be compared to find best model where agent nodes spend less energy, drop less number of packets and take less time to neutralize the intruder.

3.4 Proposed Counter Attack Models and their operation

Terminology Used:

1. **REER**: Route error. A node sends RRER if it does not have a route to destination
2. **RREQ**: Route request. A node which wants a route to the destination broadcasts RREQ
3. **RREP**: Route reply. If a node has route to the destination, it will reply to source with RREP.
4. **TTL**: Time to live. A time stamp beyond which, a packet is considered as invalid
5. **MAXTTL**: Maximum time to live. Time stamp to cover the entire network.

3.4.1 **Round Robin :**

Multiple selected agent nodes $A_i \in G_a$ (Group of agent nodes) send packets to an intruder node with a random packet size $\boldsymbol{P_i}$ for a time period $\boldsymbol{T_i}$. The value of $\boldsymbol{T_i}$ depends upon how the agent nodes are configured. If agent nodes are configured in such a way that they cannot consume more than X% of their available energy then $\boldsymbol{T_i}$ will be the time required for agent node to consume X% of its available energy. Otherwise, each agent node will select a random value for $\boldsymbol{T_i}$. The figure below shows the round robin scheme.

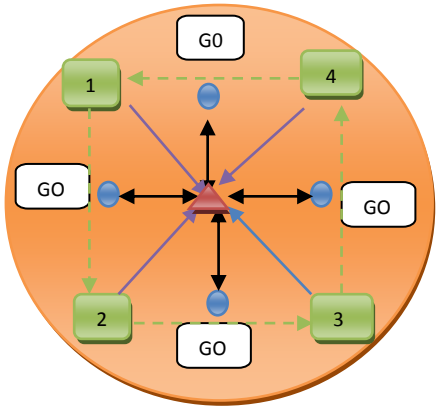
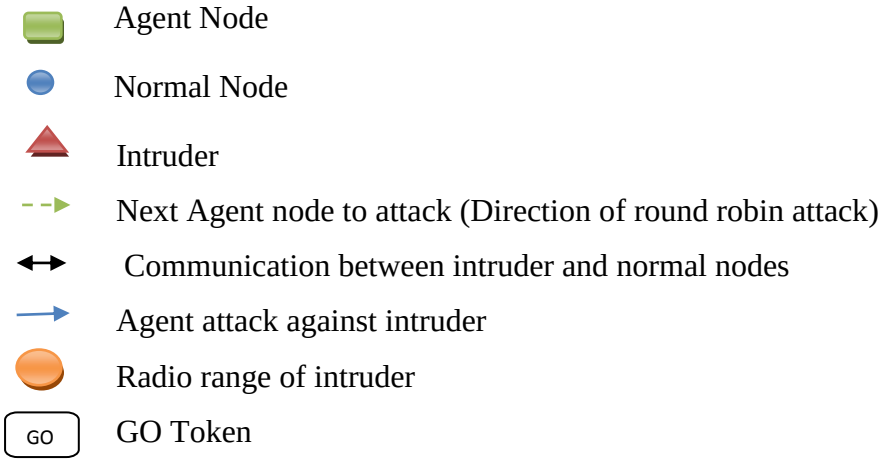


Fig 3.2 Round Robin Counter Attack Model



The thesis will not consider how an agent node identifies its immediate left and right neighbor agent nodes or how and who will generate a GO token

We make the following assumptions:

I. Each agent node has a table called "**LEFT_RIGHT_NEIGHBOUR**". This table has three fields: **LEFT_AGENT, RIGHT_AGENT, FORWARDED_TO**

Field	Meaning	Default Value
LEFT_AGENT	An agent left of this agent	NULL
RIGHT_AGENT	An agent right of this agent	NULL
FORWARDED_TO	An agent to which this agent has forwarded the token	NULL

II. Each agent node has another table called" INTRUDER_DETAILS". This table has two fields: INTRUDER_ID, STATUS

Field	Meaning	Default value
INTRUDER_ID	Node Id of Intruder	NULL
STATUS	Indicates current status of intruder, 0- alive, 1- dead	0

III. GO token has two fields

Field	Meaning	Default Value
FORWARDING_TO	An agent node to which this agent node has sent the GO token	NULL
FORWARDED_FROM	An agent node from which this agent node has received the GO token	NULL

Algorithm: Initialization // Initialize LEFT_RIGHT_NEIGHBOUR table for each agent node

Step1. **for** i=0 to |Ga| **do**, //|Ga| will give number of agent nodes

A[i].LEFT_RIGHT_NEIGHBOUR.LEFT_AGENT ← Node ID of immediate left agent

A[i]. LEFT_RIGHT_NEIGHBOUR.RIGHT_AGENT ← Node ID of immediate right agent

end

Algorithm: Round Robin // Operation of Round Robin counter-attack model

Step 1: **if**(A[i].INTRUDER_DETAILS.STATUS=1)

exit

else

goto step 2

Step2. randomly choose packet size Pi

Step3. randomly choose attack duration Ti

Step4. **while** (T_i > 0 and A[i].INTRUDER_DETAILS.STATUS=0)

do

Send packets

if(A[i] receive REER packet) **then** //If route broken

while(TTL <= MAX TTL)

do,

Broadcast RREQ with address of intruder

if (RREP not received within TTL time)

New TTL←old TTL*2

end

else

Decrement T_i ← T_i -1

Continue sending packets

end if

end

goto step 6

Step 5. Set A[i].INTRUDER_DETAILS.STATUS← 1 //1 indicates intruder is dead or out of radio range of agent A[i]

Step6. **if** (GO.FORWARDED_FROM = NULL) //A[i] is first agent to start counter attack

Set GO.FORWARDED_FROM← Node id of A[i]

Randomly pick A[i].LEFT_RIGHT_NEIGHBOUR.LEFT_AGENT A[i].LEFT_RIGHT_NEIGHBOUR.RIGHT_AGENT and forward

GO token

else if(GO.FORWARDED_FROM= A[i].LEFT_RIGHT_NEIGHBOUR.LEFT_AGENT)

Set GO.FORWARDED_FROM← Node id of A[i]

Forward GO token to A[i].LEFT_RIGHT_NEIGHBOUR.RIGHT_AGENT

else if(GO.FORWARDING_AGENT= A[i].LEFT_RIGHT_NEIGHBOUR.RIGHT_AGENT)

Set Set GO.FORWARDED_FROM← Node id of A[i]

Forward GO token to A[i].LEFT_RIGHT_NEIGHBOUR.RIGHT_AGENT

Operation:

Once the intruder identity has been confirmed by the network, the identity of the intruder will be multicast to the group of agent nodes. On receiving the intruder identity, each agent node will call *Initialization* to populate its INTRUDER_DETAILS table with INTRUDER_ID and initial STATUS as 0 indicating intruder is alive and present inside the network. The network will generate a GO token and randomly assign it to an agent node. The node which has possession of a GO token will activate a Round Robin Attack. The Agent node randomly chooses a packet size P_i and attack duration T_i. The packet size P_i and attack duration T_i for agent A_i are independent of those for agent A_{i-1}. Then agent A_i evaluates the **while** condition. If the condition evaluates to true, agent A_i will start flooding the intruder node with packets. If A_i stops hearing periodic radio beacons from the intruder, it will assume that either the intruder is dead or it is out of radio transmission range of agent A_i . It will then update the STATUS field in table INTRUDER_DETAILS and forward the GO token to the next agent. If A_i has initial possession of the token, it will have the choice to forward the token to either its immediate right agent or left agent. However, if A_i has received the token from other agent node, forwarding of token depends upon direction from which it has received the token. If A_i has received the token from left agent, it must forward the token to only its immediate right agent and vice-versa unless until its right agent or left agent value is NULL. The next agent node will again call the Round Robin Attack and each step will be executed as previously.

The attack will stop when:

- the intruder node consumes all its energy and eventually dies
- the intruder node shuts down itself
- the intruder leaves the network
- all agent nodes consumes their energy and eventually die

When an agent node sends a packet to the intruder, it awaits for an ACK from the intruder until TTL expires. If the intruder dies or shutdown itself or moves out of network, the agent node will receive RERR (route error) message. If the agent node does

not receive RREP within TTL time, it will double the TTL value and retransmit RREQ with a new TTL value. The Agent node will repeat this until the TTL value exceeds the MAX TTL value. Even at MAXTTL if an agent node does not receive an RREP packet, it will assume that intruder is either dead or shutdown itself or is out of network. It will then update the STATUS field to 1 and pass the GO token to the next agent in line. Eventually, when each agent node has STATUS value as 1, the **if** condition in step 1 will be TRUE and the attack stopped.

It is important to note that agent nodes are free to vary the packet size for each packet that they send to the intruder as well the content of packet. This will help agent nodes to probe what kind of data the intruder is interested in by recording any variation in response time from the intruder for a transmitted packet, what is intruder's processing capacity and communication bandwidth by measuring delay in the response time for a transmitted packet. This probing will be useful for models which aim to learn about the intruder's critical resources and the kinds of data it is interested in. An agent node can choose to flood the intruder node with only control packets or it may choose to embed false data in order to waste the intruder's storage buffer and increase packet processing time hence forcing the intruder to waste more energy on packet processing.

3.4.2 Flooding Attack

Multiple selected agent nodes send packets to the single intruder with a random packet size P_i and random period T_i. The purpose of this attack is to force the intruder node to decrease its communication with other ordinary nodes and eventually enter into a DoS status. The value of T_i depends upon how the agent nodes are configured. If agent nodes are configured in such a way that they cannot consume more than X% of their available energy then T_i will be the time required for an agent node to consume X% of its available energy. Otherwise, each agent node will select a random value for T_i. Figure 3.3 below shows the Flooding counter-attack model in operation.

We make the following assumptions:

I. Each agent node has a table" INTRUDER_DETAILS". This table has two field: INTRUDER_ID, STATUS

Field	Meaning	Default Value
INTRUDER_ID	Node id of intruder	NULL
STATUS	Indicate status of intruder. 0-alive, 1- dead or out of network	0

Algorithm: Flooding Counter-Attack //Algorithm to carry out Flooding attack

Step1**: if**(A[i].INTRUDER_DETAILS.STATUS=1)

exit

else

goto step 2

Step2. randomly choose packet size P_i

Step3. randomly choose attack duration T_i

Step4. **while**(T_i > 0 and A[i].INTRUDER_DETAILS.STATUS=0)

do

Send packets

if(A[i] receive REER packet) //If link broken

then

while(TTL <= MAXTTL)

do,

Broadcast RREQ with address of intruder

if(RREP not received within TTL time)

New TTL←old TTL*2

end

else

Decrement Ti ← Ti-1

Continue sending packets

end if

end

goto step 6

Step 5. Set A[i].INTRUDER_DETAILS.STATUS← 1 //1 indicates intruder is dead or out of radio range of agent A[i]

Step6. goto Step 1

Operation:

Once the intruder identity is confirmed by the network, the identity of the intruder will be multicast to the group of agent nodes. On receiving the intruder identity, each agent node will call *Initialization* to populate its INTRUDER_DETAILS table with INTRUDER_ID and initial STATUS as 0 indicating intruder is alive and present inside the network. After initialization, each agent node will call *Flooding Attack.* Each node will first check the STATUS field. The attack will happen only if STATUS filed has value set to 0. As the attack has not yet started, the condition will be evaluated to false when the agent node calls the algorithm the first time. Steps 2 to 5 will be executed and respective actions will be taken by agent nodes independently. One important point to remember here is that the flooding attack is non-cooperative. The action of one agent node is completely independent of another. At step 6, the algorithm repeats itself. The attack will continue till one of the following occurs:

- the intruder node consumes all its energy and eventually dies
- the intruder node shutdowns itself
- the intruder leaves the network
- all the agent nodes have consumed their energy and eventually die

The logic of updating the STATUS field is the same as explained at the end of the Round Robin attack.

The figure below shows the Flooding Attack model in operation.

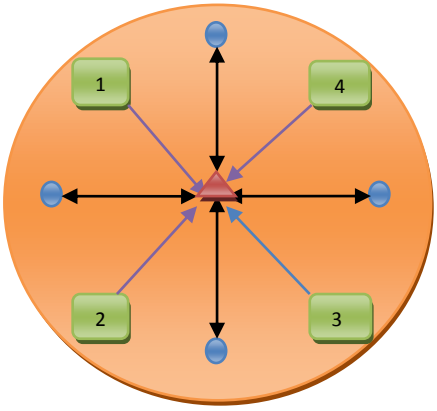


Fig 3.3 Flooding Counter Attack Model

3.4.3 Self Whisper Round Robin Attack:

Two randomly selected agent nodes A_i and A_j in G_a send packets to each other with a random packet size P_i and random sending period T_i. The communication channel between A_i and A_j will be through the intruder. The key idea is to use the intruder as a "Router". This kind of attack has two advantages over the previous two attack models where Round Robin focused on wasting intruder energy and Flooding focused on overpowering the intruder and forcing it to enter DoS mode. The self whisper model on the other hand focuses on both, that is, wasting intruder energy as well as its communication bandwidth. To achieve this, multiple pairs of communications between agents will happen simultaneously.

If there are *N* agents around the intruder then we have $FLOOR(\left|\frac{N}{2}\right|)$ pairs of communication. All of these communications go through intruder. For example, for *N*=5, we have FLOOR (5/2) =2, that is, maximum of 2 communication channels are opened through the intruder.

We make the following assumptions:

I. Each agent node has a table called "**LEFT_RIGHT_NEIGHBOUR**". This table has three fields: **LEFT_AGENT, RIGHT_AGENT, FORWARDED_TO**

Field	Meaning	Default Value
LEFT_AGENT	An agent node left to this agent node	NULL
RIGHT_AGENT	An agent node right to this agent node	NULL
FORWARDED_TO	An agent node to which this agent has forwarded the GO token	NULL

II. Each agent node has a table "**GROUP**". The table has two fields: **NODE_ ID**, **BUSY_STATUS** field will take only two values, 0 indicating not busy and 1 indicating busy.

Field	Meaning	Default Value
NODE_ID	Node id of agents	NULL

BUSY_STATUS	If agent node is currently active in counter-attack	NULL

III. Each agent node has another table" **INTRUDER_DETAILS**". This table has two fields: **INTRUDER_ID, STATUS**

Field	Meaning	Default Value
INTRUDER_ID	Node id of intruder	NULL
STATUS	Indicate status of intruder. 0-alive, 1- dead or out of network	0

IV. **GO** token has two fields "**FORWARDED_FROM**" and "**FORWARDED_TO**". Value for these fields will be set to NULL at the time of generating the GO token.

Field	Meaning	Default Value
FORWARDING_TO	An agent node to which this agent node has send the GO token	NULL
FORWARDED_FROM	An agent node from which this agent node has received the GO token	NULL

Algorithm: Initialization // Initialize LEFT_RIGHT_NEIGHBOUR table and GROUP Table

Step1. **for** i=0 to |Ga|

do, //|Ga| will give number of agent nodes

A[i].LEFT_RIGHT_NEIGHBOUR.LEFT_AGENT ← NODE_ ID of immediate left agent

A[i]. LEFT_RIGHT_NEIGHBOUR RIGHT_AGENT ← NODE_ ID of immediate right agent

for j=0 to |G_a|

do,

A[i].GROUP.NODE_ID[j]←A[j]

A[i].GROUP.BUSY_STATUS[j]←0 //0 indicates node is not busy

end

end

Algorithm: Self Whisper Round Robin Counter-Attack // Algorithm for self whisper counter-attack

Step 1: **if**(A[i].INTRUDER_DETAILS.STATUS=1)
exit
else
goto step 2

Step2. randomly select an agent A[j]from GROUP table with BUSY_STATUS=0
Step3. Send invitation to A[j] for communication
Step4. **If** invitation rejected Set A[j].BUSY_STATUS←1
goto step 2
else
goto step 5
Step5. Randomly choose packet size Pi
Step6. Randomly choose attack duration Ti
Step4. **while**(Ti>0 and A[i].INTRUDER_DETAILS.STATUS=0)
do
Send packets
if(A[i] receive REER packet) //If link broken
then
while(TTL <= MAXTTL)
do,
Broadcast RREQ with address of intruder
if(RREP not received within TTL time)
New TTL←old TTL*2
end
else
Decrement Ti ← Ti-1
Continue sending packets
end if
end
goto step 6
Step 5. Set A[i].INTRUDER_DETAILS.STATUS← 1 //1 indicates intruder is dead or out of radio range of agent A[i]
Step6. **if** (GO.FORWARDED_FROM= NULL) //Ai is first agent to start counter attack
Set GO. FORWARDED_FROM ← NODE_ID of A[i]

Randomly pick A[i].LEFT_RIGHT_NEIGHBOUR.LEFT_AGENT or
A[i].LEFT_RIGHT_NEIGHBOUR.RIGHT_AGENT and forward
GO token
else if(GO.FORWARDED_FROM= A[i].LEFT_RIGHT_NEIGHBOUR.LEFT_AGENT)
Set GO.FORWARDED_FROM← NODE_ID of A[i]
Forward GO token to A[i].LEFT_RIGHT_NEIGHBOUR.RIGHT_AGENT
else if(GO.FORWARDIED_FROM= A[i].LEFT_RIGHT_NEIGHBOUR.RIGHT_AGENT)
Set GO.FORWARDED_FROM← NODE_ID of A[i]
Forward GO token to A[i].LEFT_RIGHT_NEIGHBOUR.RIGHT_AGENT

Algorithm: Reply Invitation //Algorithm explains how an agent node will reply to invitation send by another agent node during counter-attack

Step1**.** Receive Invitation
Step2**. If** currently holding GO token then
Reject invitation
else if currently communicating with another agent node
Reject invitation
else
Accept invitation

Operation:

Upon receiving the intruder identity from the network, all agent nodes will call *Initialization.* This will populate the table LEFT_RIGHT_NEIGHBOUR with the immediately left agent NODE_ID and immediately right agent NODE_ID. The INTRUDER_DETAILS table gets populated and initial STATUS is set to 0 indicating that intruder is active. Each agent node will also populate the GROUP table with NODE_ID of the rest of the agents as well as its own NODE_ID . The BUSY_STATUS field is set to 0 indicating that the other agent nodes are free. After initialization, the two agent nodes which have possession of the GO token will randomly pick an agent node

with STATUS as 0 and will try to establish a communication with it. If the other agent is either busy or is a holder of GO token, it will reject the invitation. To reject an invitation, when an agent node receive the invitation or request for communication, the recipient agent will call *Reply invitation* method. If it is currently having a possession of a GO token or is currently communicating with another agent node, it will reject the invitation, otherwise it will accept the invitation and a communication channel will be established between the transmit agent node and recipient agent node. To make sure that communications between two agents nodes run through the intruder, during the path establishment phase, the node which sends the invitation to another agent node, will accept only those RREP which results in a path from the initiator agent node to recipient agent node through intruder node. The thesis will not discuss about any modifications that needs to be made on top of the AODV protocol since there are many mechanisms available to implement it.

Another important point to note is that during the first round of the counter attack, except the two agent nodes which initiate the counter attack, all other nodes must pass the GO token to an agent node in the direction of the GO token. If an agent node receives a GO token from another agent node to its left, it can only pass the GO token to another agent to its right. This constraint will make sure that the counter attack takes place in a round robin fashion. The attack will stop once the intruder STATUS is set to 1 in the INTRUDER_DETAILS table inside each agent node. Figure 3.4 below shows the Self Whisper Round Robin Attack Model in operation.

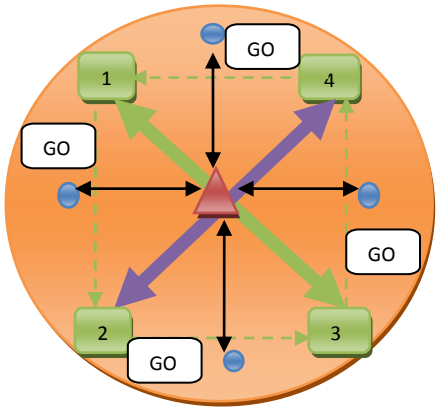


Fig 3.4 Self Whisper Round Robin Counter Attack Model

CHAPTER IV

SIMULATION

4.1 Objective of Simulation

The objective of the simulation is to study the effectiveness of the proposed counter attack models against a single intruder and test our hypothesis as listed below for each proposed model. Through simulation we study the following characteristics of each model as the number of agent nodes is increased:

- Energy consumption of intruder.
- Time taken to neutralize/marginalize the intruder
- Number of packets dropped by intruder

Our hypothesis is:

- For the Round Robin counter-attack model, an increase in the number. of agent nodes should not have any impact on packets dropped by the intruder because at any given time, only one of the agent nodes will be attacking the intruder Hence, an increase in the number of agent nodes in the counter-attack should result in an increase in average energy consumption rate for each agent node as newly added agent node will also consume energy during the attack, resulting in an increase in overall energy consumption by a group of agents.
- For the Flooding counter-attack model, an increase in the number of agent nodes should result in an increase in packet loss by the intruder. This work will determine the maximum threshold of the number of agent nodes beyond which,

increasing the number of agent nodes joining the counter attack, will not result in an increase in the packet loss rate by the intruder.

- For the Self Whisper Round Robin attack model, our hypothesis is that as the number. of agent nodes increase, the overall time to consume the intruder's communication bandwidth and energy should decrease.

4.2 Simulation Tool

We use NS-2 as our simulation software. It is free, open source software under the GUI license [11]. NS-2 provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks. The five different ad-hoc routing protocols currently implemented for mobile networking in NS-2 are DSDV, DSR, AODV, TORA and PUMA. NS-2 has very strong support for wireless networking and many ad hoc networking protocols have been implemented in NS-2. However, the visualization of NS-2 is not that good as compared to other available simulators like Net Logo.
References to NS-2 are listed in the reference section of this thesis.

4.3 Simulation Model

Our models have three main objects or components: Ordinary nodes, an attacker and a group of agent nodes. We assume in all our models that the attacker has already initiated the attack by diverting some, if not all of the network traffic through itself. To accomplish it, the attacker can use false route information and broadcast it to ordinary nodes.

4.3.1 Base Model (Why we need it?)

The objective of having a base model as shown in figure 3.4 below is to measure two important parameters that will be used in our three counter-attack models. These two parameters are:

I. To measure energy consumption rate (Joule/sec) of a non-intruder node as a function of packet rate (packets/sec).

II. To find the threshold packet rate beyond which, increase in packet rate has little or no effect on the intruder.

The first parameter will be required to compute quantum time for the Round Robin and Self Whisper counter-attack models. If the network manager decides that an agent node should spend only X% of its initial energy during each turn that it will get during round robin and self whisper attacks, the first parameter (energy consumption rate) can be used to compute the *quantum* (time slot) as follows:

Let's assume that measured value energy consumption rate for non-intruder node is Y Joule/sec. Now let's assume that a network manager wants each agent node to spend only X Joule of energy during each turn that it will get. Then

$$Quantum = Ceil[X/Y]$$

This quantum will be used, every time an agent node receives a chance to counter-attack the intruder.

The second parameter will be used to determine the threshold *traffic* (Packet Rate) beyond which, any increase in traffic has little or no impact on intruder in terms of packet drop rate. Threshold traffic is directly proportional to buffer size of intruder. In our experiments, we use the threshold traffic as produced by measuring the 2nd parameter mentioned above for a fixed buffer size and do not change it.

We have designed a base model with two ordinary nodes communicating with each other through an intruder. Ordinary nodes are unaware of the presence of the intruder. Four possible energy consumption states are identified: transmitting, receiving, idle and sleep. The first two states represent when the node is transmitting and receiving packets respectively, the idle state is a state when the node is neither transmitting nor receiving packets, but actively listening to the radio transceiver and hence consuming energy. The sleep state is a very low power consumption state where the node can neither receive nor transmit packets. The cost associated with each packet at a node is represented as the total of incremental cost *m* (unit cost/byte) proportional to the packet *size* and a fixed cost *b* associated with channel acquisition:

$$Cost = m * size + b$$

NS-2 has this energy model inbuilt and during the simulation setup, the energy consumption rate for each of the four states as mentioned above is set. All nodes have the same communication bandwidth (2Mbps), same initial energy (100 Joule) and the same MAC interface queue length of 50 packets of size 512 bytes. All the nodes are running on the same wireless card i.e. *LUCENT IEEE 802.11 2 MBPS WAVELAN PC CARD 2.4 GHZ DIRECT SEQUENCE SPREAD SPECTRUM* [14]. Freenay et. al. [*14*] and Allard, et. al [*15*] have investigated energy consumption of network interface cards. The above mentioned lucent wireless card has been investigated by Freenay [14] for its energy consumption under four states, namely, transmitting, receiving, and idle and sleep.

The base model is showen below:

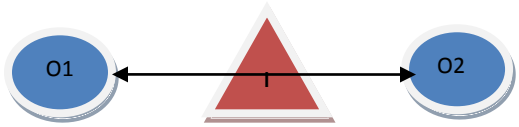


Fig 4.1 Base Model setup

O1: Ordinary Node 1

O2: Ordinary Node 2

I: Intruder

⟷ : Communication through Intruder I

4.4 Simulation Setup

Simulation duration: 600 sec

Attack starts: 5 sec after simulation starts

The main parameters used for node creation is mentioned below:

Parameter	Value	Used For
Transmission Range	250 m	Controls the transmission

		range of a node
RTS/CTS packet threshold energy for acceptance in MAC layer	3.65262e-10 joule	Any packet has RSSI value less then this will be dropped.
IFQ length	50	Node buffer size at MAC layer
MAC type	802_11	Accessing the physical layer carrier
Wireless Interface card	Lucent 2.5 GHz DSSS silver card with 2Mbps speed	Wireless Communication between nodes
Transmission Power	1.3 Joule/sec	Power consumed per sec during transmission
Receive Power	0.9 Joule/sec	Power consumed per sec during receiving
Idle Power	0.2 Joule/sec	Power consumed per sec during node being idle
Packet Size	512 bytes	Size of data packet

Table 4.1 Parameters for node configuration in NS-2

More information on creating mobile nodes, topology and generating traffic, see [13], [16] and [17].

4.5 TCP vs. UDP

Three counter-attack models are simulated for both TCP as well as UDP based agents. The discussion below is primarily oriented from a NS-2 point of view.

4.5.1 TCP (Transmission Control Protocol): - TCP [18] has several objectives:

- Adapt the transmission rate of packets to available bandwidth.
- Avoid congestion at the network.
- Create a reliable connection by re-transmitting lost packets.

In order to control the transmission rate, the number of packets that not yet been received is bounded by a parameter called W***indow (W)***. This means that the source is obliged to wait and stop transmission. The number of packets that it had transmitted and that has not been "acknowledged" by receiver reached W. TCP uses ***Dynamic Congestion Window.***

The basic idea is as follows: When the window is small, it can grow rapidly, when it reaches the large value, it can only grow slowly. When congestion is detected, window size decreases drastically. This dynamic behavior of TCP allows rapid adaptation to congestion and uses network bandwidth efficiently.

More precisely, let's assume W_t is our initial estimation of network bandwidth and W is our current window size. The window W starts with initial value 1. For each received ACKs, the W is incremented by 1. So when we receive 1st ACK, W= W+1=2. This phase is called "slow start". The W will keep continuing increasing in this fashion till $W = W_t$.

Next, we enter the second phase called the "Congestion avoidance' phase, where W increases by FLOOR (W/2) for each received ACK. After transmitting W packets, W increases by 1. If we transmit W packets at time t, then at time W+RTT, we transmit W+ 1 packet, at time t+2RTT, we transmit W+2 packets. Hence once can see that window growth is linear. Reference [18] gives more details about TCP.

4.5.2 User Datagram Protocol

User Datagram Protocol (UDP) [19] can send messages, referred to as *datagram's,* to other hosts on an Internet Protocol (IP) network without requiring prior communications to set up special transmission channels or data paths. UDP uses a simple transmission model without implicit hand-shaking dialogues for providing reliability, ordering, or data integrity. Thus, UDP provides an unreliable service and datagram's may arrive out of order, appear duplicated, or go missing without notice. UDP assumes that error checking and correction is either not necessary or performed in the application, avoiding the overhead of such processing at the network interface level. Time-sensitive applications often use UDP because dropping packets is preferable to waiting for delayed packets, which may not be an option in a real-time system. Most often, UDP applications do not employ reliability mechanisms. Unlike TCP, UDP based applications don't necessarily have congestion avoidance and control mechanisms. Lacking reliability, UDP applications must generally be willing to accept some loss, errors or duplication. Reference [18] gives more details about UDP.

4.6 Experimental Results

Objective:

I. To measure the energy consumption rate of the intruder as well as the cumulative and individual energy consumption rates of agent nodes as a function of the number of packets/sec and number of agent nodes.

II. To measure packet drop rate of intruder as well as cumulative and individual packet drop rate (packets/sec) of agent nodes as a function of the number of packets/sec and number of agent nodes.

III. To find threshold value for packet rate (packets/sec) beyond which, increase in packet rate by agent nodes has little or no effect on results.

IV. To measure the time taken by each counter-attack model to consume all the energy of intruder, that is, time $\boldsymbol{T}$ when intruder's energy $\boldsymbol{I_e}$ =0, as a function of packet rate (packets/sec) and number of agent nodes.

4.6.1: Base Model

Simulation Setup:

Parameter	**Value**
Number of ordinary nodes	2
Number of agent nodes	0
Number of Intruder	1
Routing Protocol	AODV
Radio transmission range	250 m
Initial Energy	100 joule for each node
Packet Rate	2,4,8,16,32,64,128,256,512,1024

Table 4.2 Common node configuration parameter for Base Model

Hypothesis:

1. Ordinary Node#1 generates the traffic, and ordinary node Node#2 receives the traffic. Intruder receives packet from ordinary O1and then forwards the packet to ordinary O2. Because the intruder node is acting both as a receiver and as a

transmitter, energy consumption rate of intruder should be higher than that of both the ordinary nodes, O1 and O2.

2. As traffic increases, we should observe higher packet drop rates at the intruder. The packet drop rate of intruder should be higher or at least equal to packet drop rate of O1.
3. As traffic increases, the time taken for intruder to consume all its energy should decrease.

The graph below confirms our hypothesis.

O1 = Ordinary node #1

O2 = Ordinary node #2

I = Intruder

I_e = Intruder's Energy

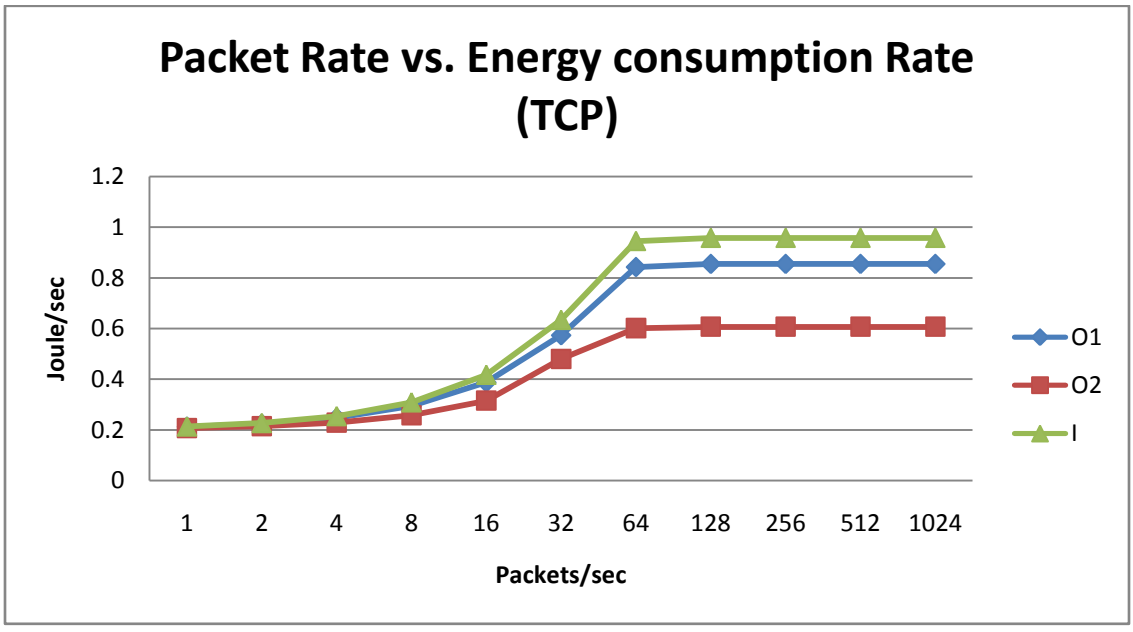


Fig 4.2 Base Model Energy consumption rate (Joule/sec) for TCP

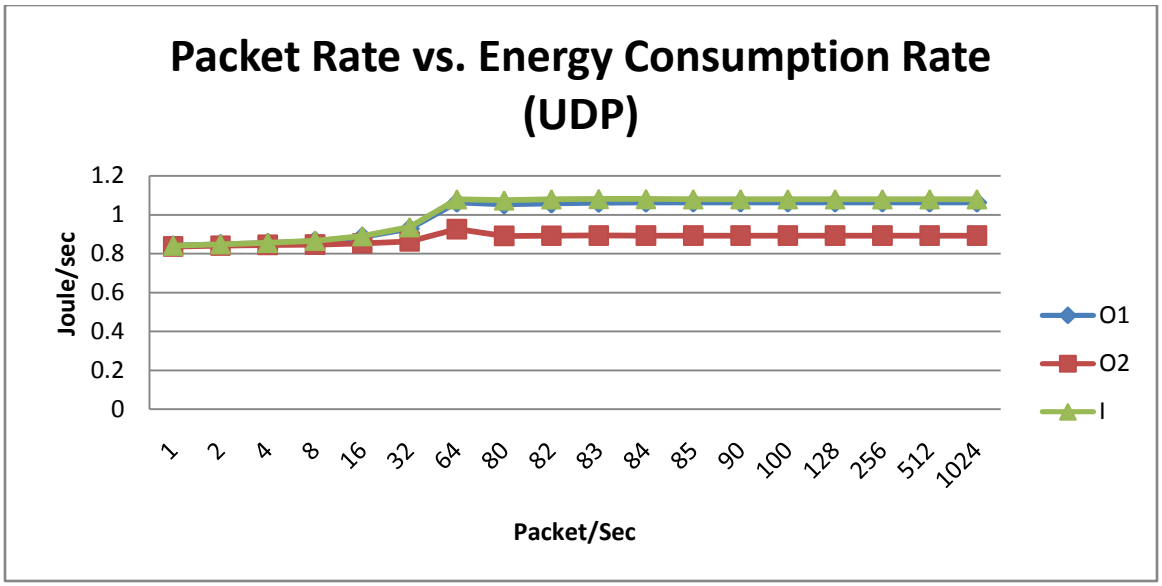


Fig 4.3 Base Model Energy consumption rate (Joule/sec) for UDP

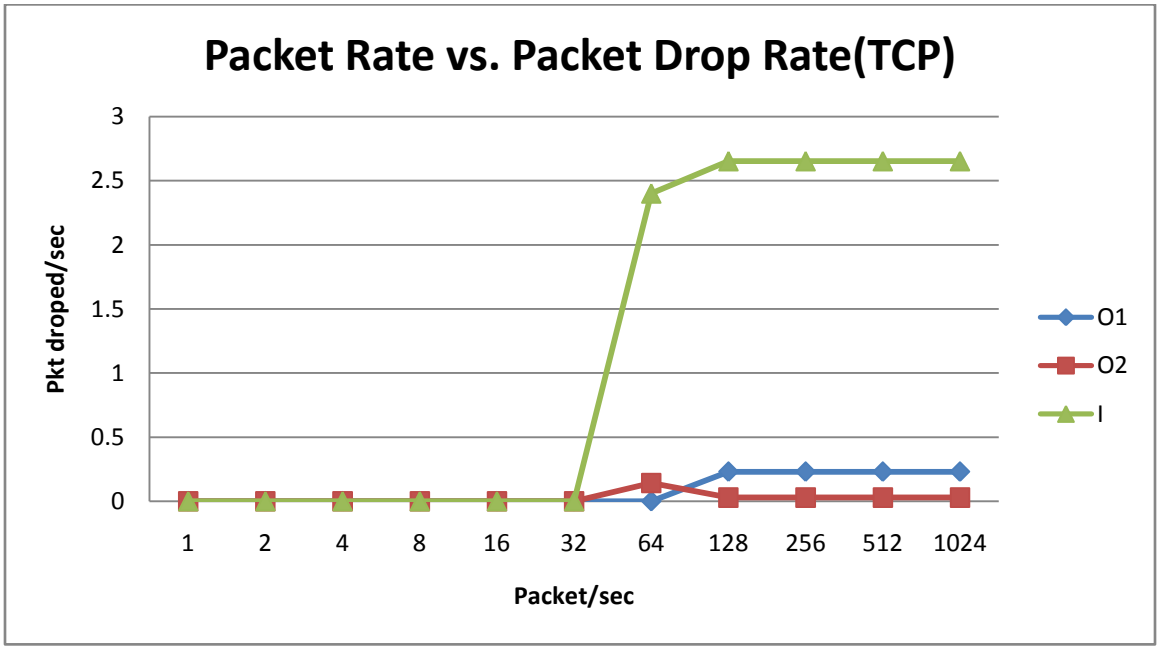


Fig 4.4 Base Model Packet Drop Rate (Packets/sec) for TCP

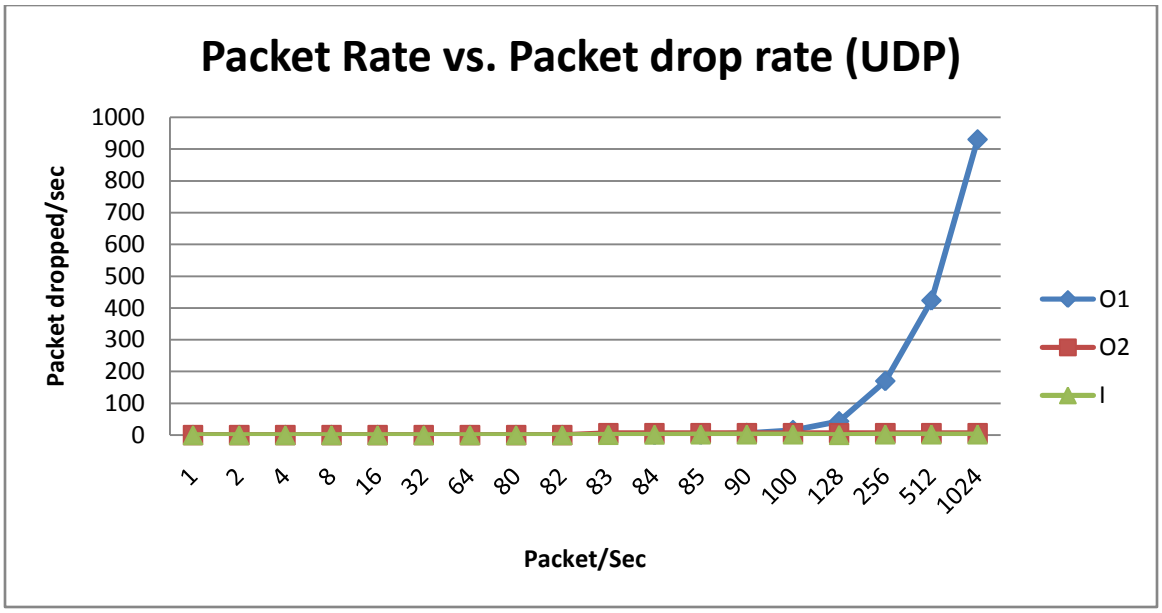


Fig 4.5 Base Model Packet Drop Rate (Packets/sec) for UDP

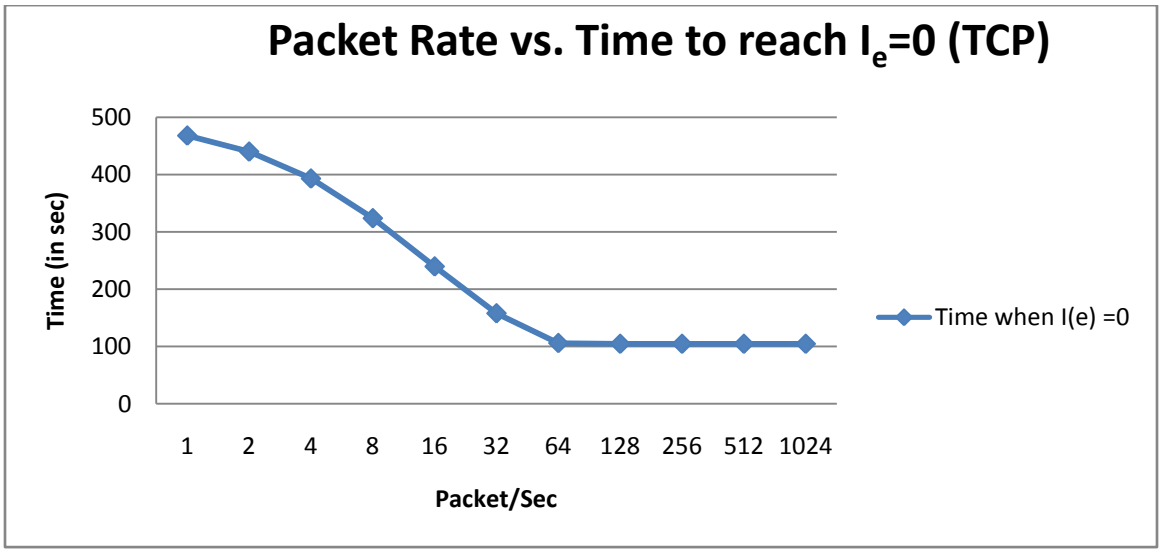


Fig 4.6 Base Model Time to reach I_e=0 for TCP

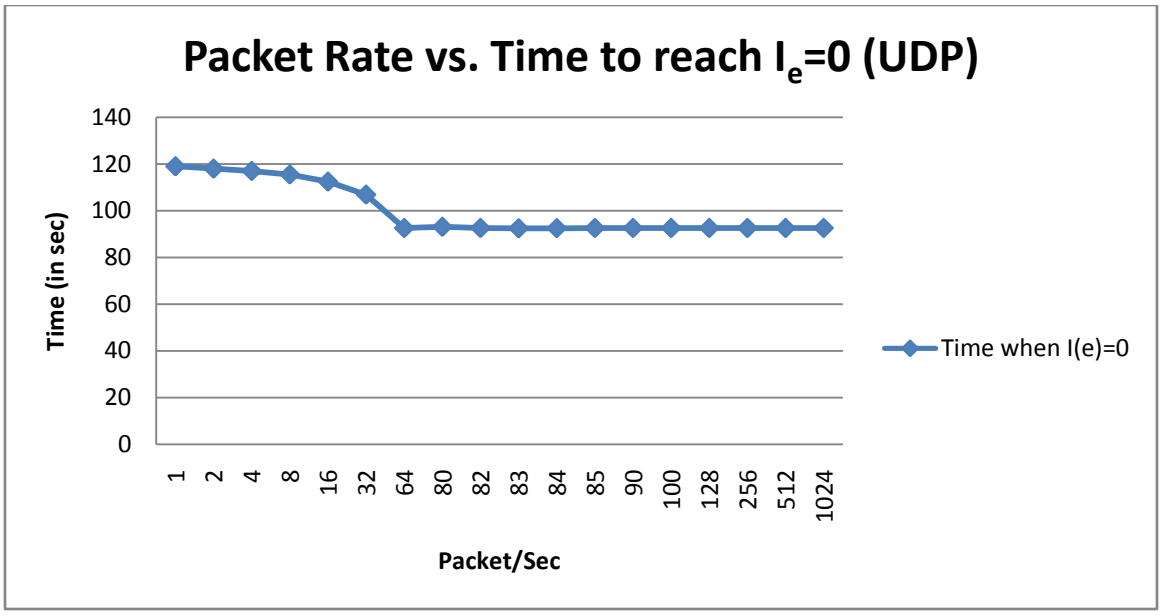


Fig 4.7 Base Model Time when I_e =0 for UDP

From the above three graphs, we can see that there is very little or no effect on energy consumption rate, packet drop rate and time taken to consume intruder's energy after the packet rate crosses 64 packets/sec in both TCP as well as in UDP simulation. This behavior is expected as each node has IFQ (At MAC layer) length of 50. Thus, once applications start generating packets in excess of 50 packets, the intruder is not able to handle such traffic and hence number of packets dropped at intruder becomes steady. The simulation shows that the threshold packet rate that an intruder can handle is 64 packets/sec of size 512 bytes for a single communication channel running through it.

However, when compared to TCP, in UDP the energy consumption rate of ordinary node O1 is almost equal to that of Intruder. As UDP does not have a congestion control mechanism, O1 keeps generating and transmitting data packets irrespective of whether Intruder (I) is capable of handling such traffic or not Traffic generation consumes energy. Hence, the more O1 generates traffic, the more it consumes energy. However, as we can see in the graphs above (figure 4.10 and figure 4.11), after the threshold packet rate is reached, the energy consumption rate of all three nodes. O1, O2 and I become steady. This is because at agent O1 the application layer is generating traffic in excess of what O1's MAC layer can handle, the excess packets are buffered into O1 buffer. If the

number of packets buffered at O1 is in excess of 50 which is the size of O1's buffer, the extra packets will be dropped by the MAC layer of O1 before it transmitted. Hence, above the threshold packet rate, the amount of traffic that the intruder receives from O1 is constant. Hence, energy consumption rate of intruder I also become steady in the case of UDP as the packet rate increases, after 100 packets/sec, the packet dropped by O1 increases rapidly because the size of IFQ length is 50 and O1 generate traffic in excess of what its MAC layer can handle. The Intruder can also handle 50 packets /sec (its buffer size). Hence, the excessive packets been generated by O1 cause a lot of collision at MAC layer. Resulting in higher packet drop rate at O1 compared to intruder.

The time taken when intruder consumes all its energy is higher in TCP then in UDP. The reason behind it is TCP has a dynamic window control mechanism to handle congestion. Whenever the intruder node is getting overloaded, it results in congestion in the link between O1 and intruder. At this time, intruder publishes a window size of 0 to O1 indicating that it cannot handle any data packet at this point of time. During this time, source O1 does not transmit any data packet. Hence, this results in a longer time period for intruder to consume all its energy.

4.6.2: Round Robin

Simulation Setup:

Parameter	Value
Number of ordinary nodes	2
Number of agent nodes	2,4,6
Number of Intruder	1
Routing Protocol	AODV
Radio transmission range	250 m
Initial Energy	100 joule for each node
Packet Rate	32,64,128,256

Table 4.3 Common node configuration parameter for Round Robin

Terminology Used in Graphs:

Agent (#2): 2 agents in the simulation

Agent (#4): 4 agents in the simulation

Agent (#6): 6 agents in the simulation

Intruder (#2): 2 agents

Intruder (#4): 4 agents

Intruder (#6): 6 agents

I_e = Intruder's Energy

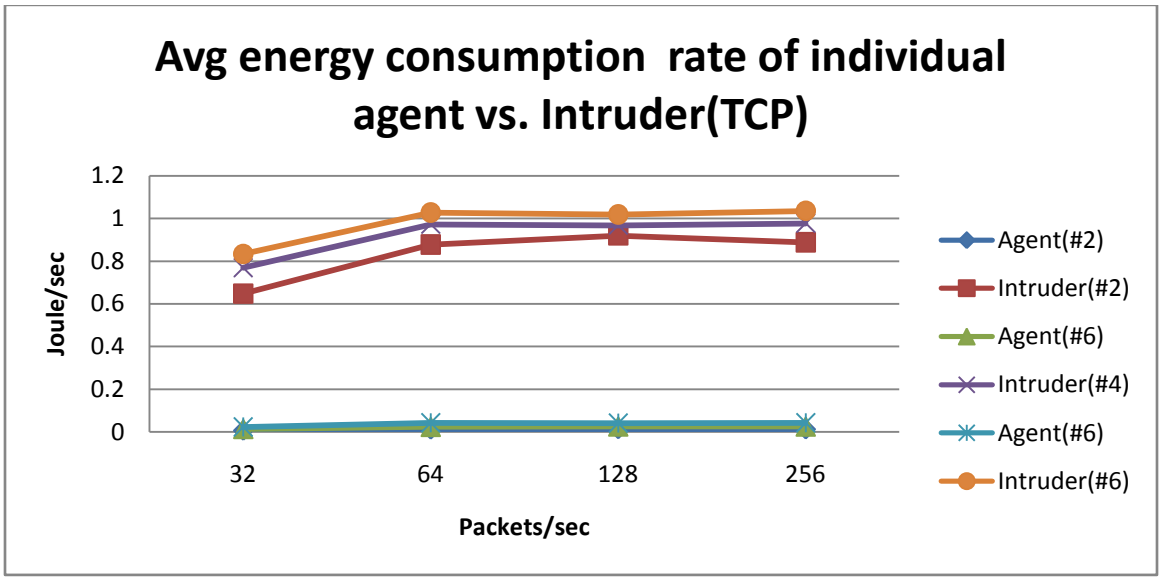


Fig 4.8 Round Robin Average energy consumption of Agents and Intruder TCP

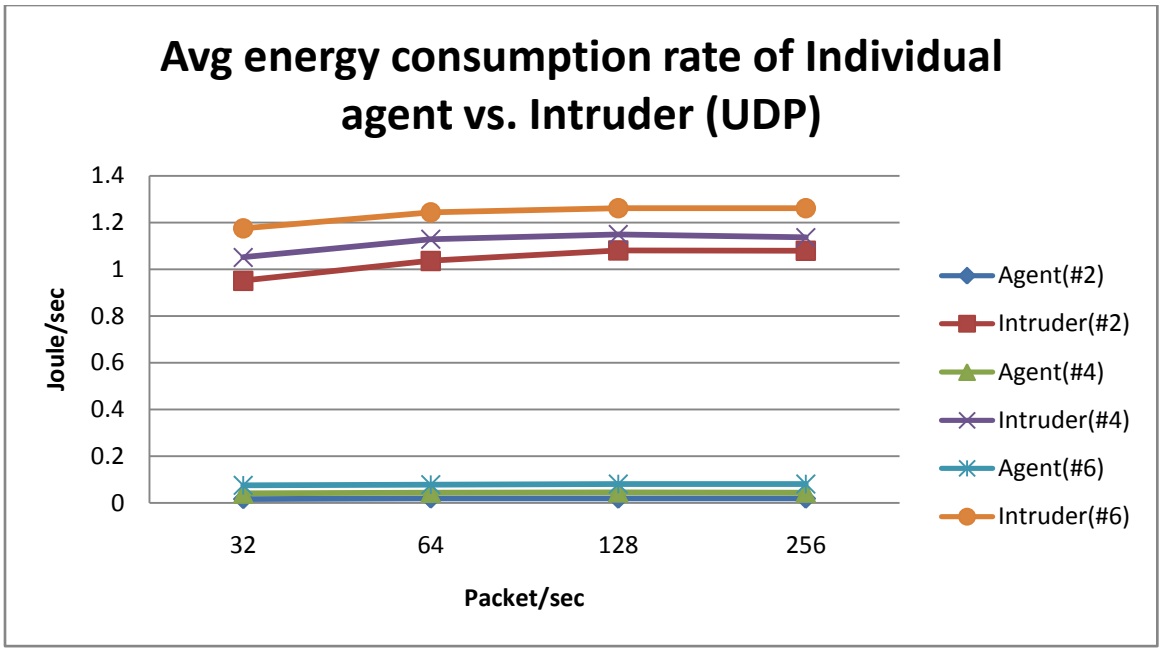


Fig 4.9 Round Robin Average energy consumption of Agents and Intruder UDP

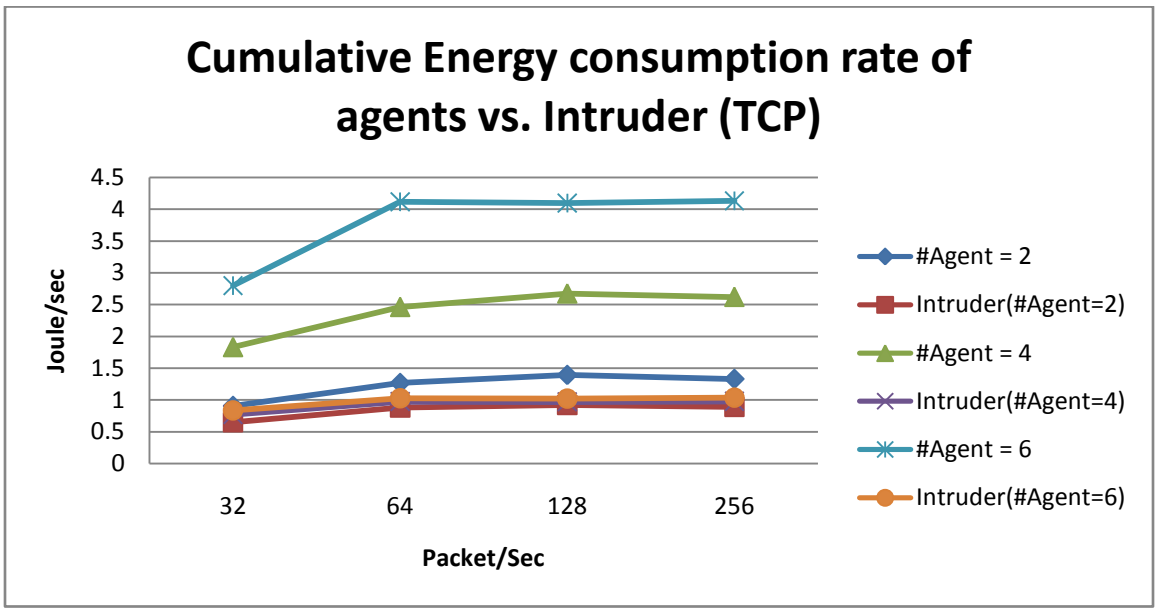


Fig 4.10 Round Robin cumulative energy consumption of Agents and Intruder TCP

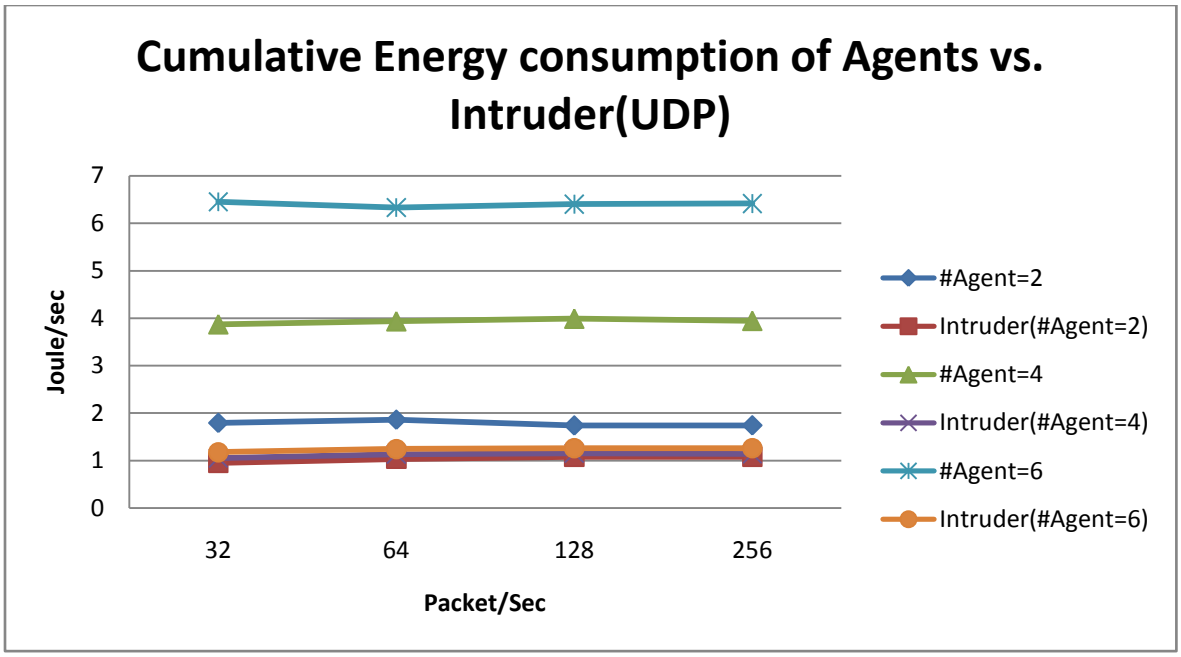


Fig 4.11 Round Robin cumulative energy consumption of Agents and Intruder UDP

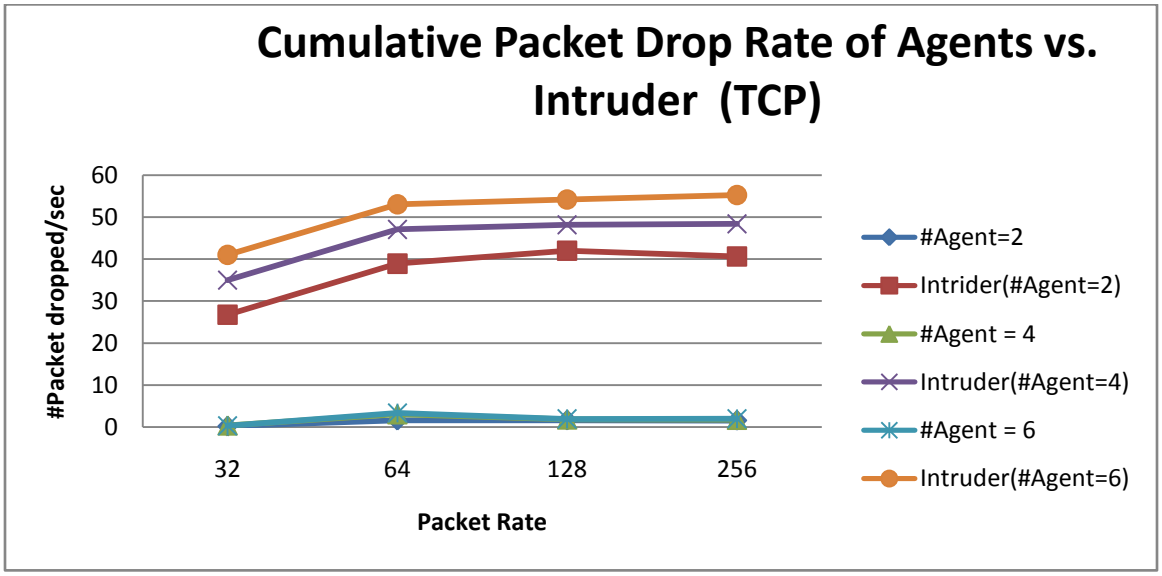


Fig 4.12 Round Robin cumulative packet drop rate of Agents and Intruder TCP

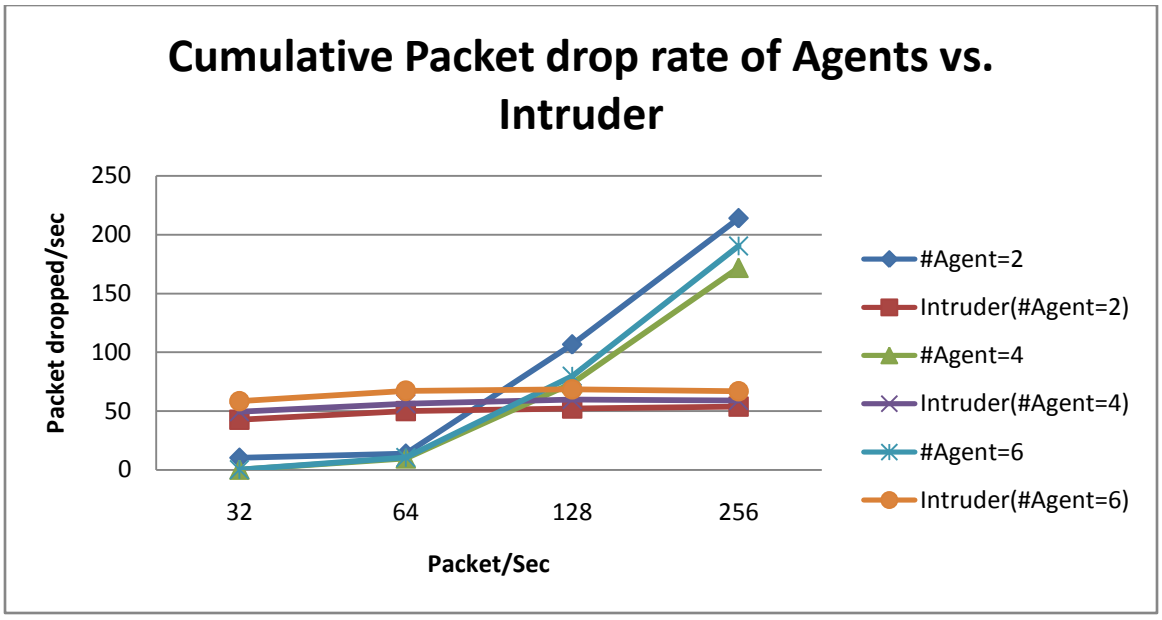


Fig 4.13 Round Robin cumulative packet drop rate of Agents and Intruder UDP

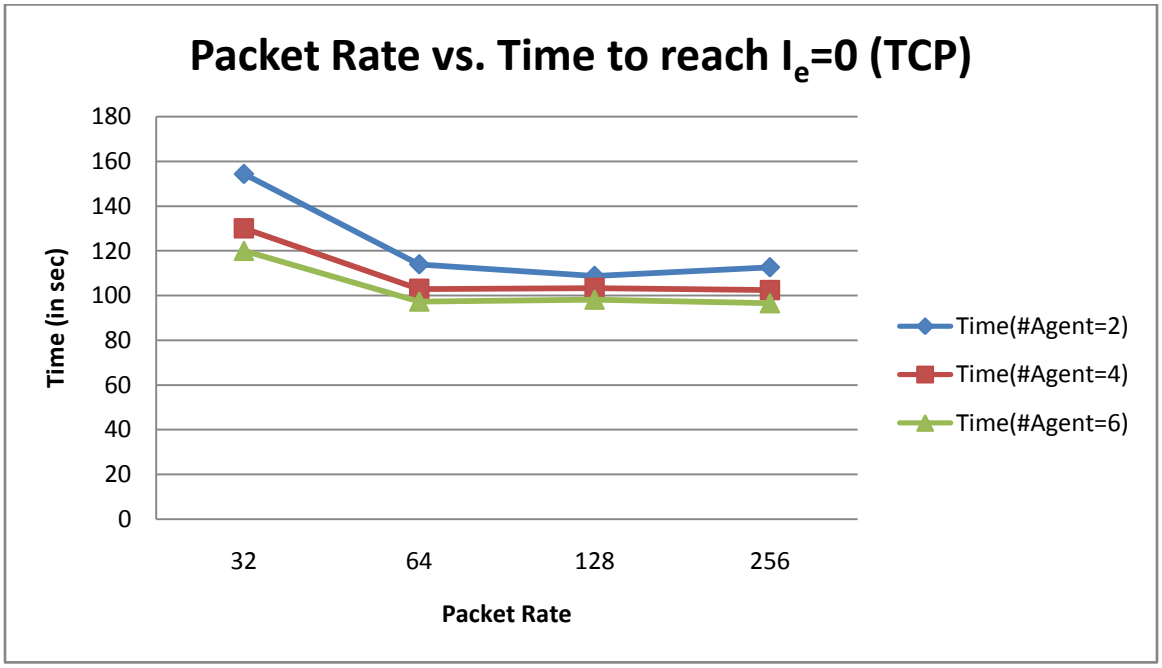


Fig 4.14 Round Robin- Effect of packet rate and #Agents on Time taken when I_e=0, TCP

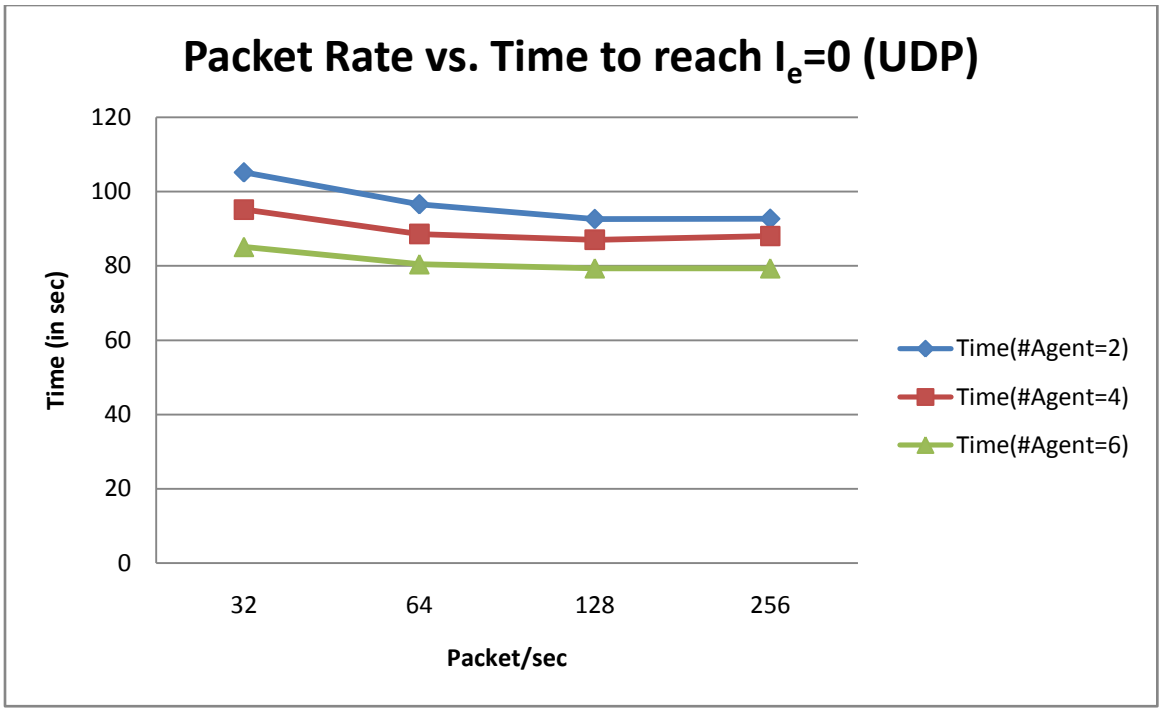


Fig 4.15 Round Robin- Effect of packet rate and #Agents on Time taken when I_e =0, UDP

As expected, when we increase the number of agent nodes, the average energy consumption rate per individual agent node should be less then energy consumption rate of Intruder. This is confirmed by Fig. 4.12 for TCP and Fig. 4.13 for UDP.

Fig 4.16 suggests that as number of agent nodes increase, the packet drop rate at intruder increases. However, once we cross threshold packet rate which is 64 packets/sec, we should be expecting a smooth curve with little or no variation. Fig 4.16 again confirms this assumption.

But the same assumption cannot be applied when we use UDP instead of TCP. As UDP is a connectionless protocol, an increase in packet rate beyond threshold packet rate will increase the packet drop by agents because agents are generating packets in excess of what their MAC layer buffer can handle which is 50 packets. Fig. 4.17 confirms this.

An increase in the number of agent nodes should result in a decrease in the time required for the intruder to consume all its energy because intruder receives more and more traffic from agent nodes. Figure 4.18 and figure 4.19 confirm the same.

Finally, The TCP version of Round Robin counter-attack model performs better than its UDP counterpart when it comes to packet drop rate. UDP version performs better when it comes to energy consumption rate at Intruder and the time taken to exhaust all the energy of Intruder.

4.6.3: Flooding

Simulation Setup:

Parameter	**Value**
Number of ordinary nodes	2
Number of agent nodes	2,4,6
Number of Intruder	1
Routing Protocol	AODV
Radio transmission range	250
Initial Energy	100 for each node
Packet Rate	32,64,128,256

Table 4.4 Common node configuration parameter for Flooding

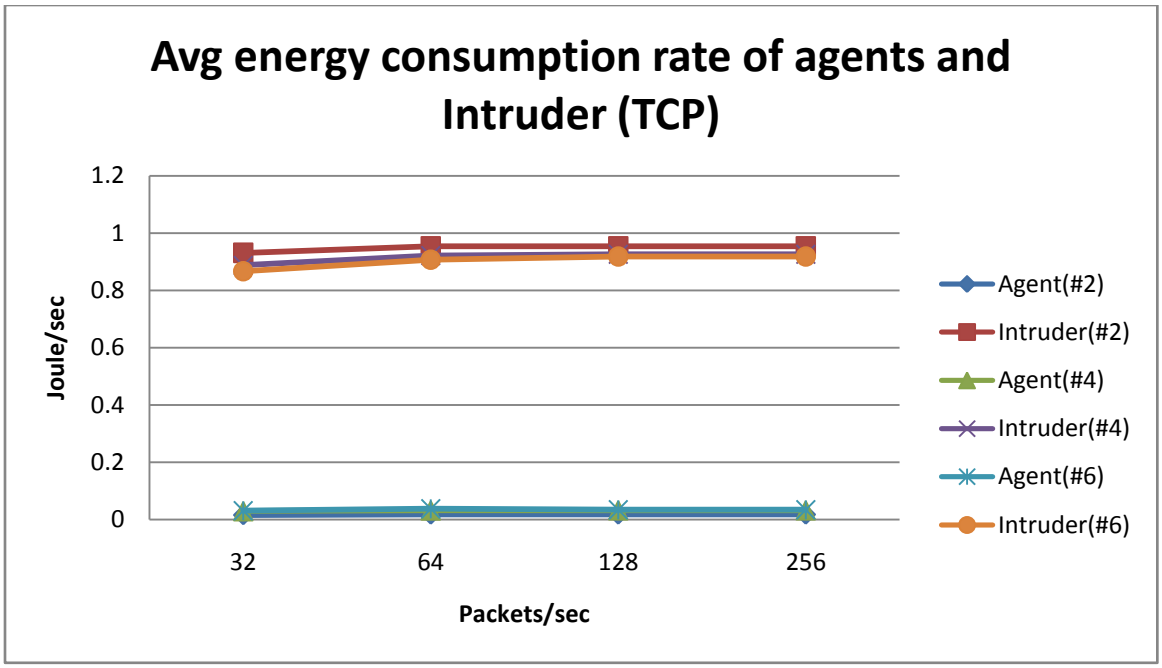


Fig 4.16 Flooding, Individual energy consumption rate of agents vs. Intruder, TCP

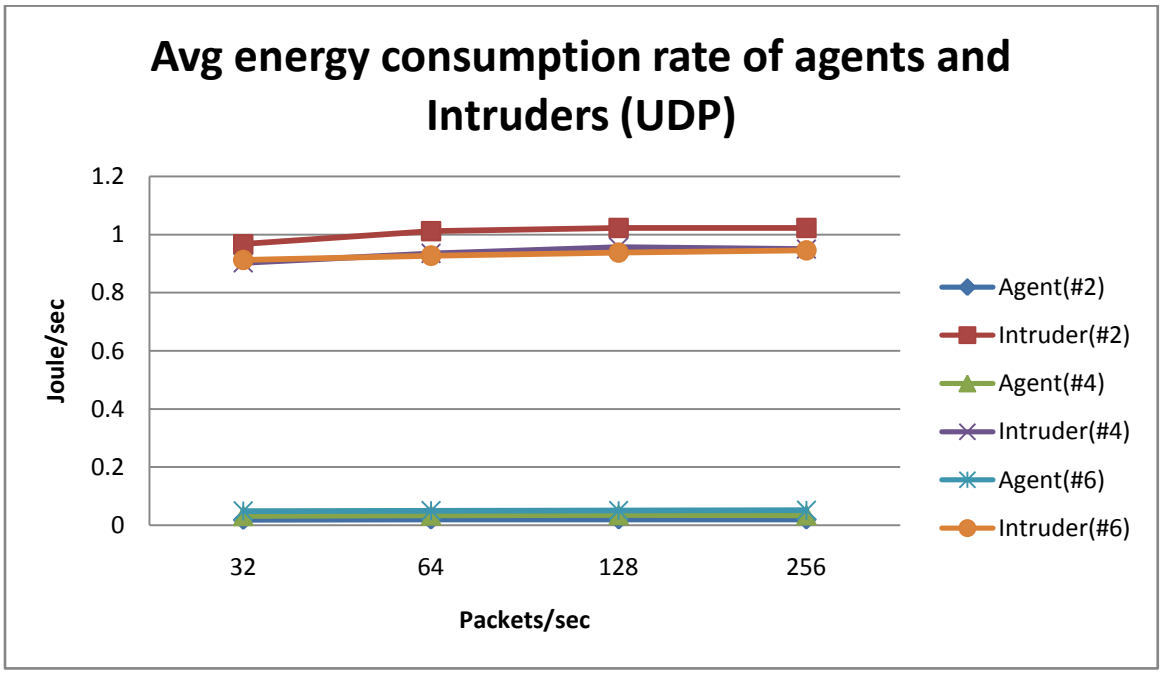


Fig 4.17 Flooding, Individual energy consumption rate of agents vs. Intruder, UDP

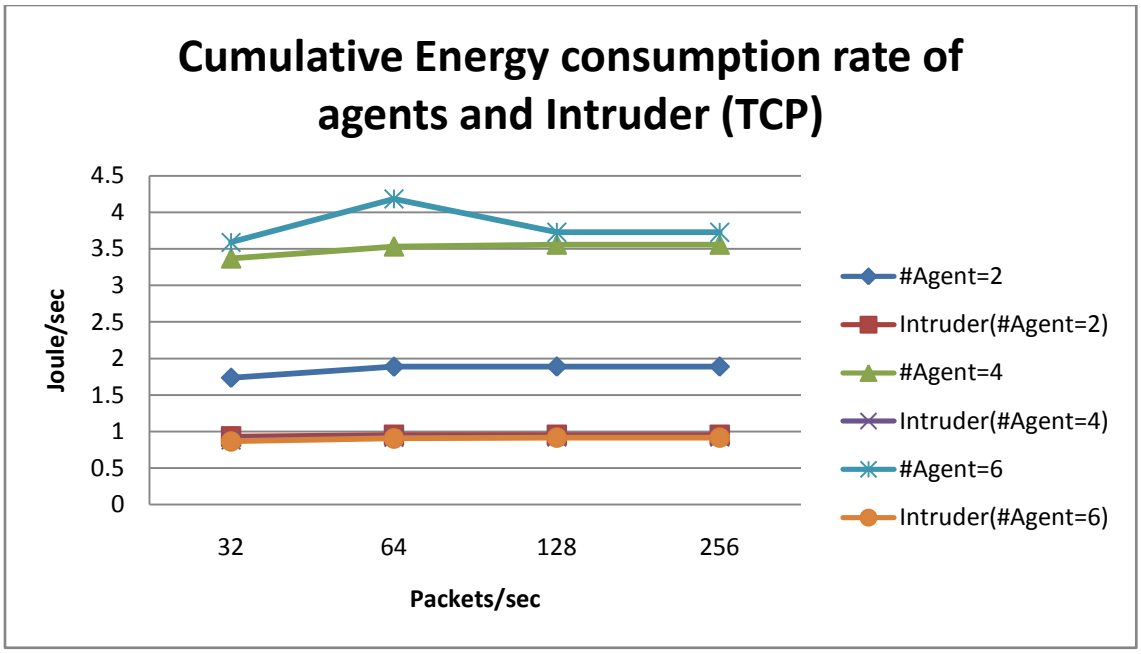


Fig 4.18 Flooding, Cumulative energy consumption rate of agents vs. Intruder, TCP

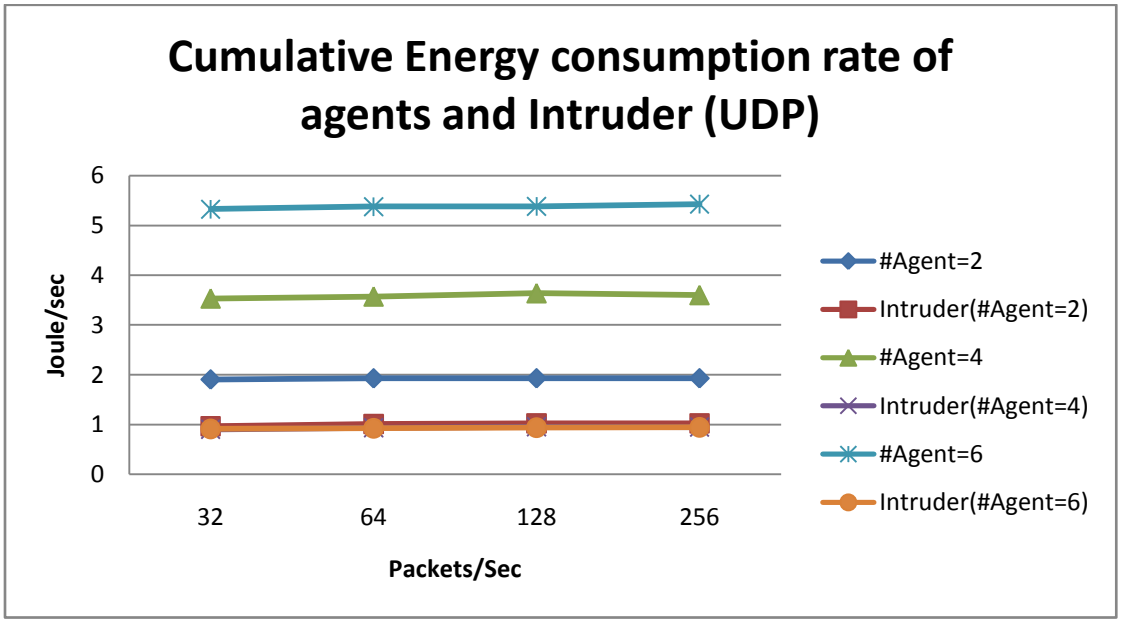


Fig 4.19 Flooding, Cumulative energy consumption rate of agents vs. Intruder, UDP

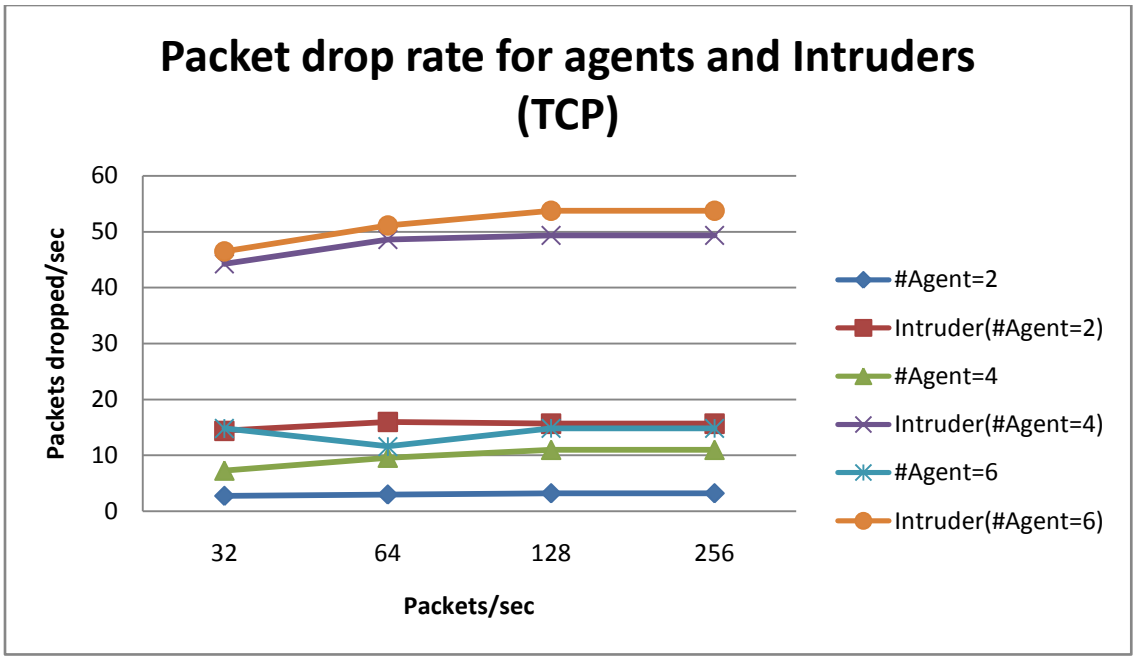


Fig 4.20 Flooding, Packet drop rate agents vs. Intruder, TCP

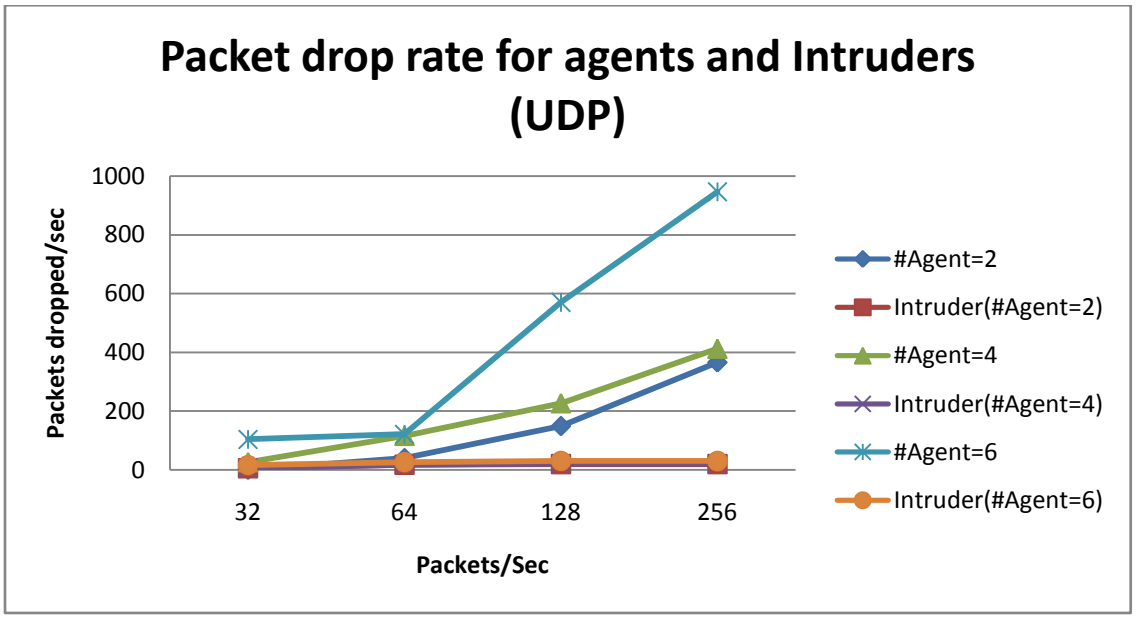


Fig 4.21 Flooding, Packet drop rate agents vs. Intruder, UDP

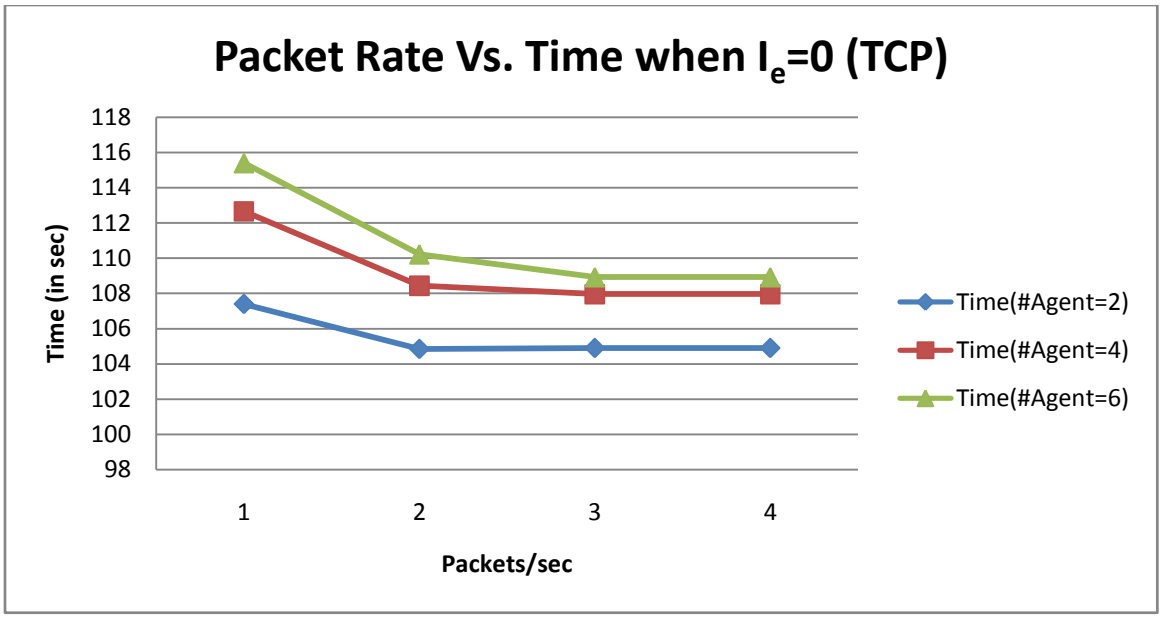


Fig 4.22 Flooding, Time taken for I_e =0, TCP

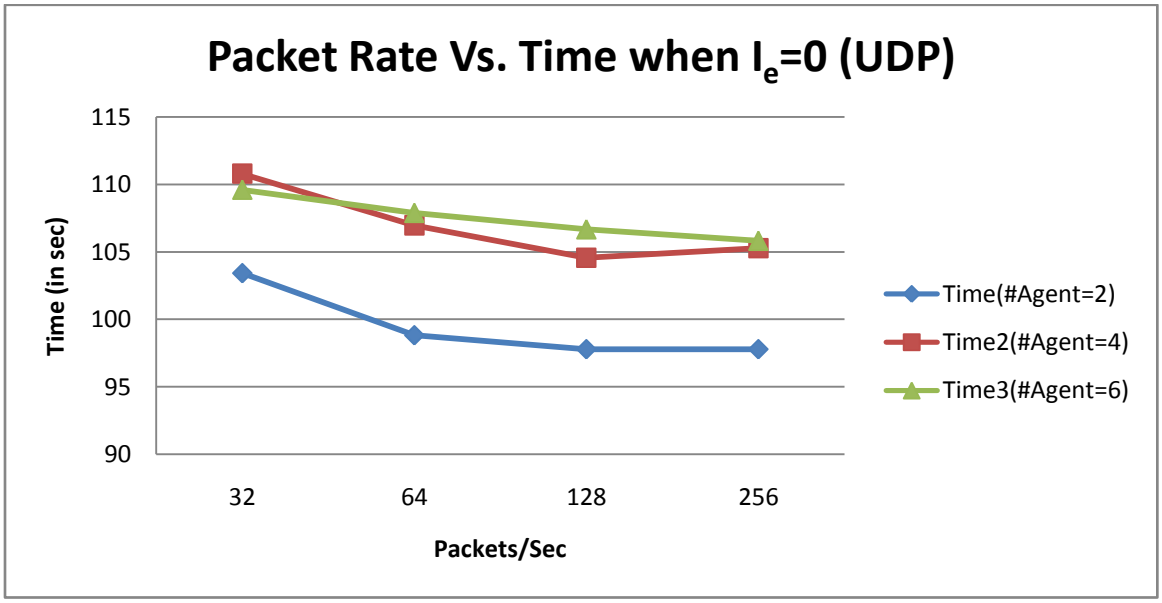


Fig 4.23 Flooding, Time taken for I_e =0, UDP

For energy consumption rate of individual agent nodes, we can see from fig 4.20 and Fig 4.21 that there is not much difference between TCP and UDP. However, UDP performs better than TCP for time taken to exhaust intruder energy (I_e). TCP version performs better when packet drop rate for intruder is the main objective.

On other hand, Fig 4.26 suggests that a smaller agent group yields lower time bound compared to a larger agent group. In the flooding counter-attack model, all agent nodes start attacking at the same time. This results in very heavy traffic in the links between agent nodes and intruder. This results in frequent publishing of window size of zero to agent nodes, resulting in frequent halt to the attack. This is confirmed by the simulations.

4.6.4: Self Whisper

Simulation Setup:

Parameter	**Value**
Number of ordinary nodes	2
Number of agent nodes	2,4,6
Number of Intruder	1
Routing Protocol	AODV
Radio transmission range	250
Initial Energy	100 for each node
Packet Rate	32,64,128,256

Table 4.5 Common node configuration parameter for Self Whisper

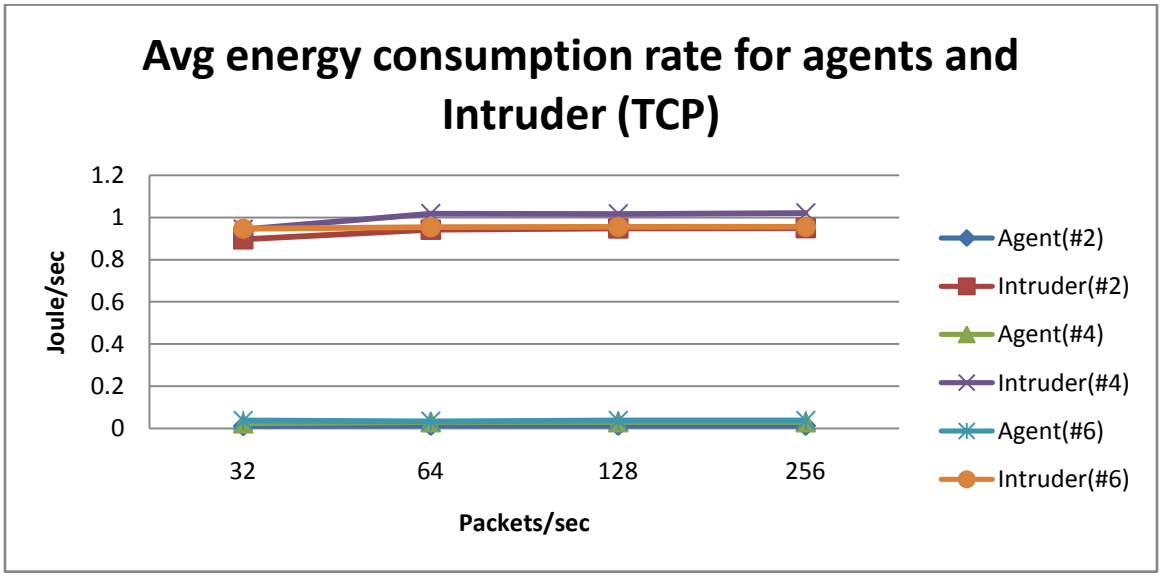


Fig 4.24 self whispers, Average energy consumption rate of Agents vs. Intruder, TCP

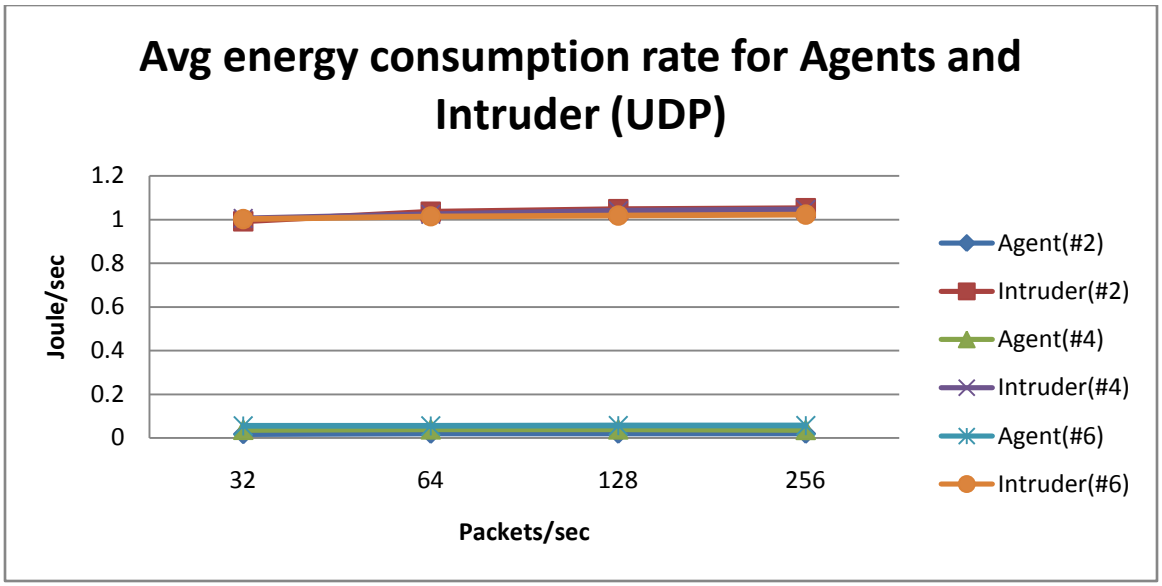


Fig 4.25 self whispers, Average energy consumption rate of Agents vs. Intruder, UDP

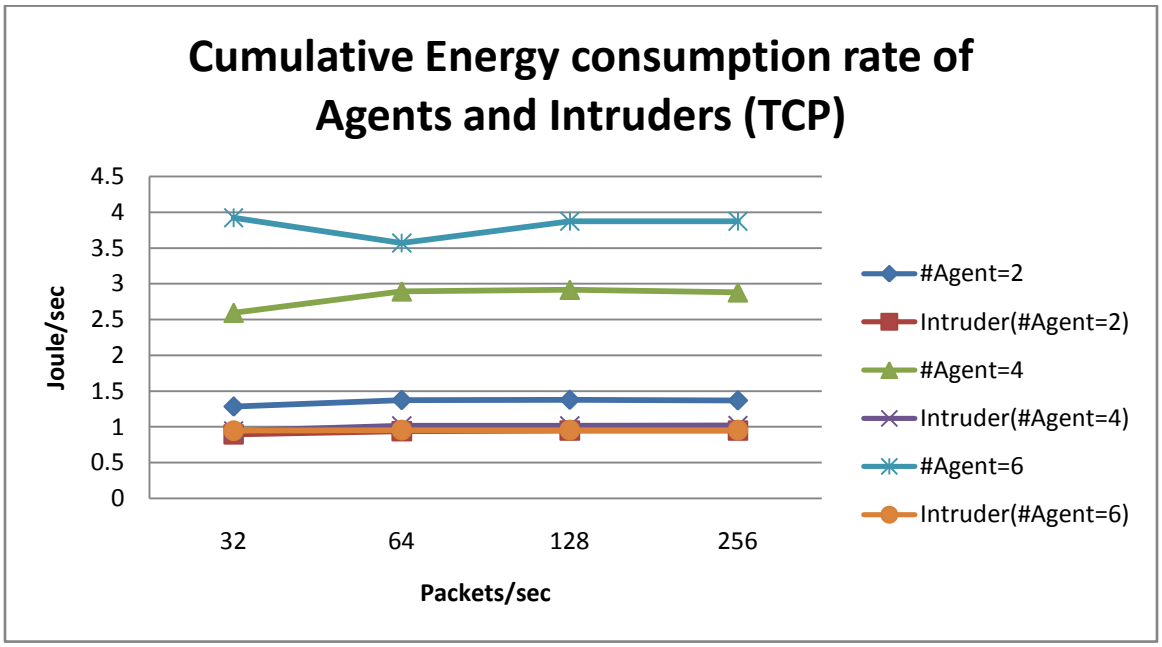


Fig 4.26 self whispers, Cumulative energy consumption rate of Agents vs. Intruder, TCP

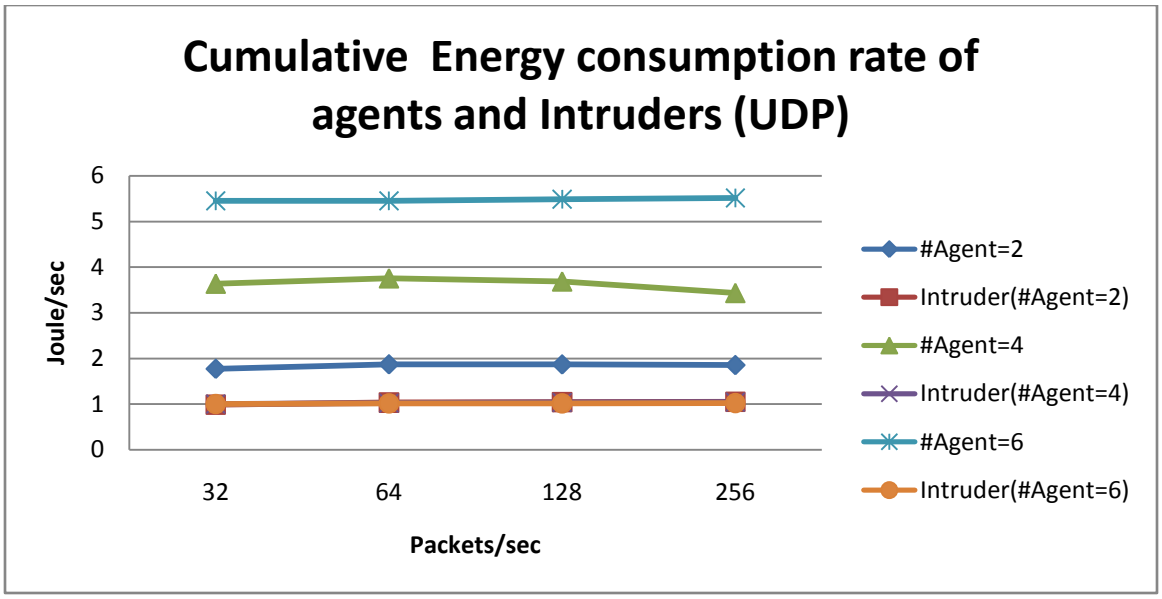


Fig 4.27 self whispers, Cumulative energy consumption rate of Agents vs. Intruder, UDP

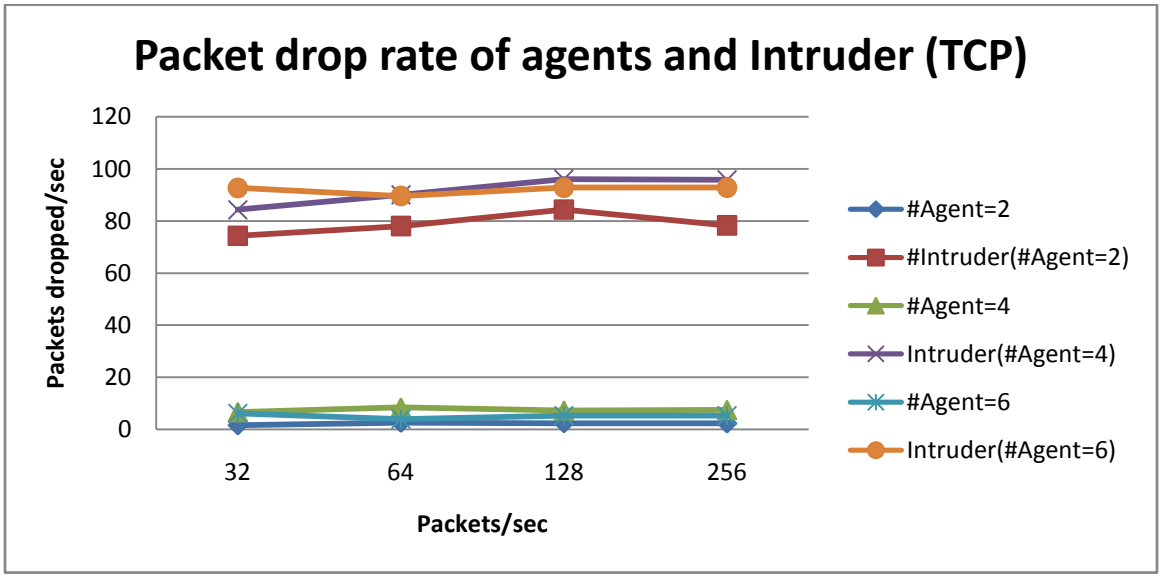


Fig 4.28 self whispers, packet drop rate of Agents vs. Intruder, TCP

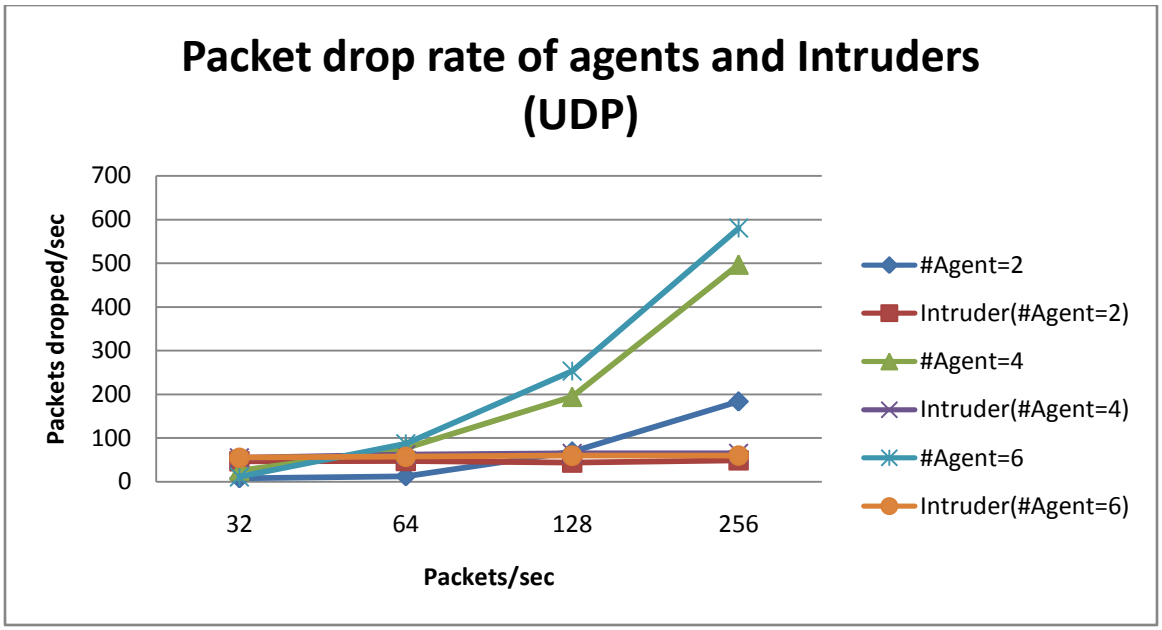


Fig 4.29 self whispers, packet drop rate of Agents vs. Intruder, UDP

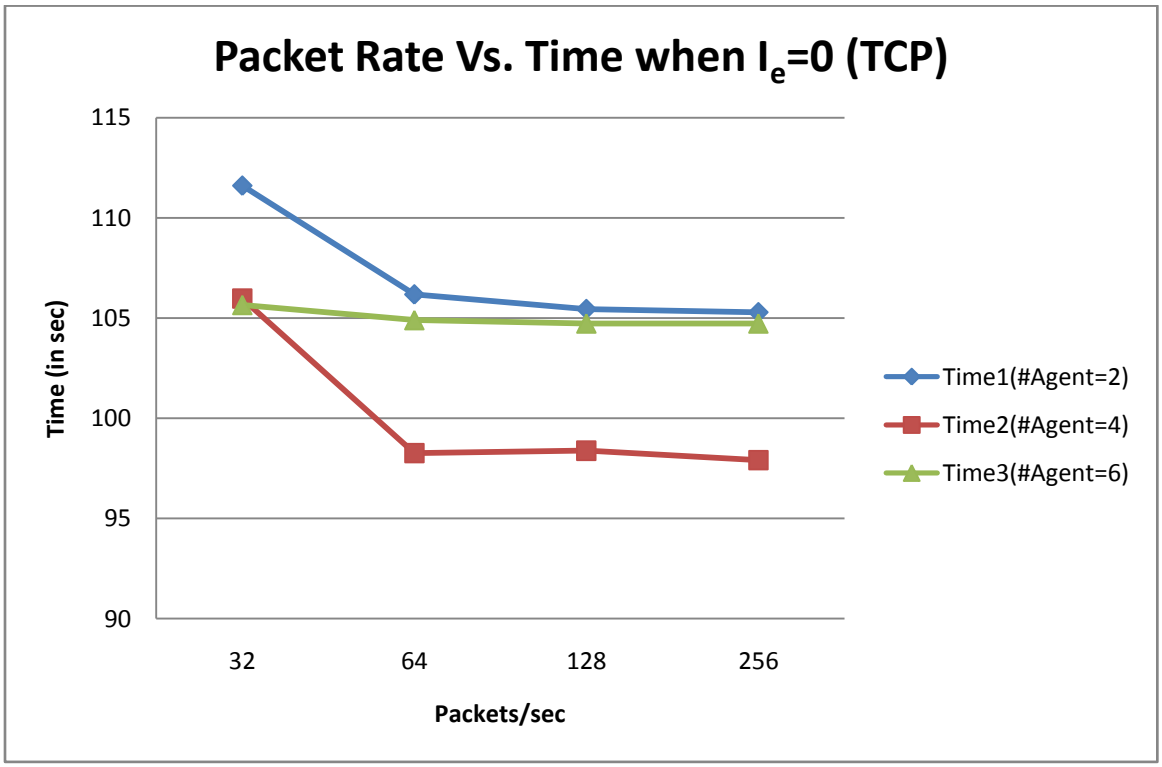


Fig 4.30 self whispers, Packet rate vs. time when I_e=0, TCP

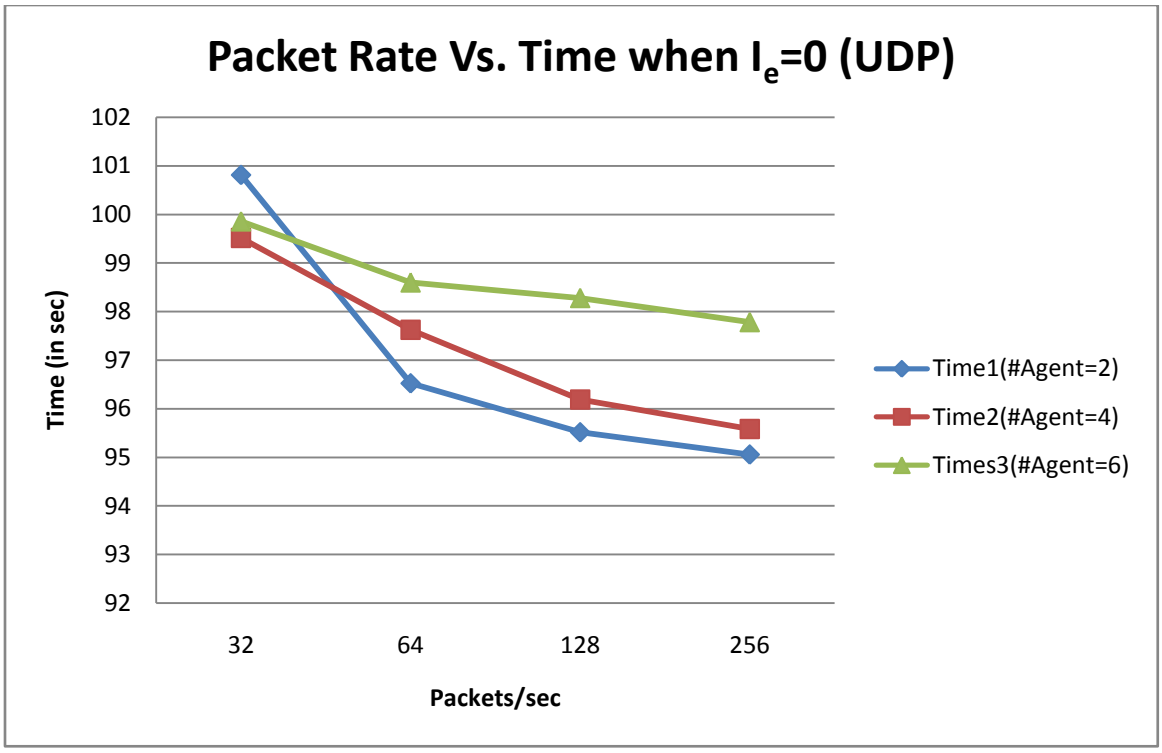


Fig 4.31 self whispers, Packet rate vs. time when I_e =0, UDP

The average energy consumption rate for agents in both TCP and UDP are almost similar. However, when it comes to packet drop rate, TCP as usual performs better then UDP which is blind to the problem of congestion in link. For the time taken by agents to exhaust the energy of Intruder, Fig. 4.34 suggest that in TCP, an increase in the number of agent nodes need not guarantee lower time bound on time needed to exhaust energy of intruder. This is confirmed from Fig 4.35.

4.7 Model Comparison

Objective: To find a counter-attack model where cost of counter-attack (in terms of energy, communication) is minimized but impact on intruder is maximized.

First, we compare TCP version of all three models, that is, Round Robin, Flooding and Self Whisper to find a model that minimizes the following parameters:

1. Cumulative energy consumption rate
2. Packet drop rate
3. Time taken to exhaust energy of intruder

for agents but maximize them for Intruders except for the 3rd parameter, where the goal is to minimize the parameter.

Secondly, we will repeat the above steps for the UDP version of all three counter attack models.

1. Comparison for lower cumulative energy consumption rate for Agents

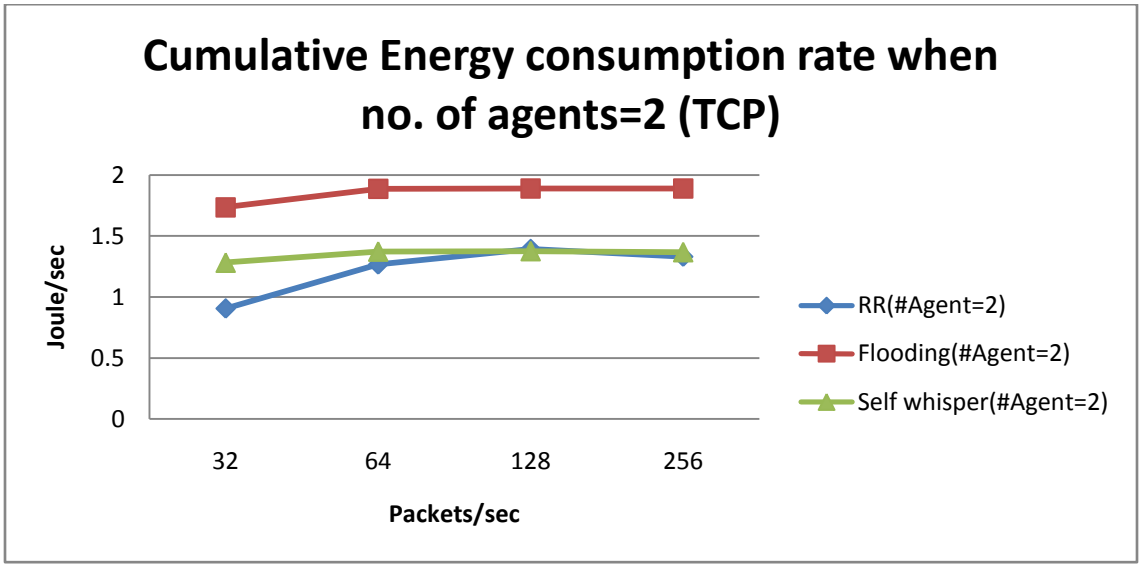


Fig 4.32 Cumulative energy consumption rate when no. of agents=2

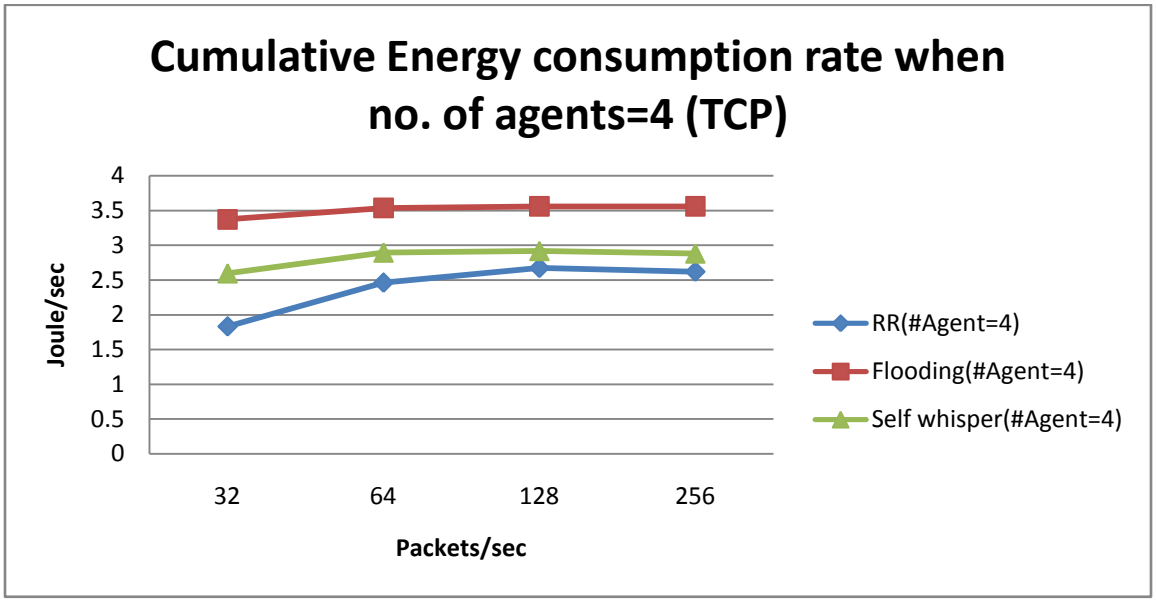


Fig 4.33 Cumulative energy consumption rate when no. of agents=4

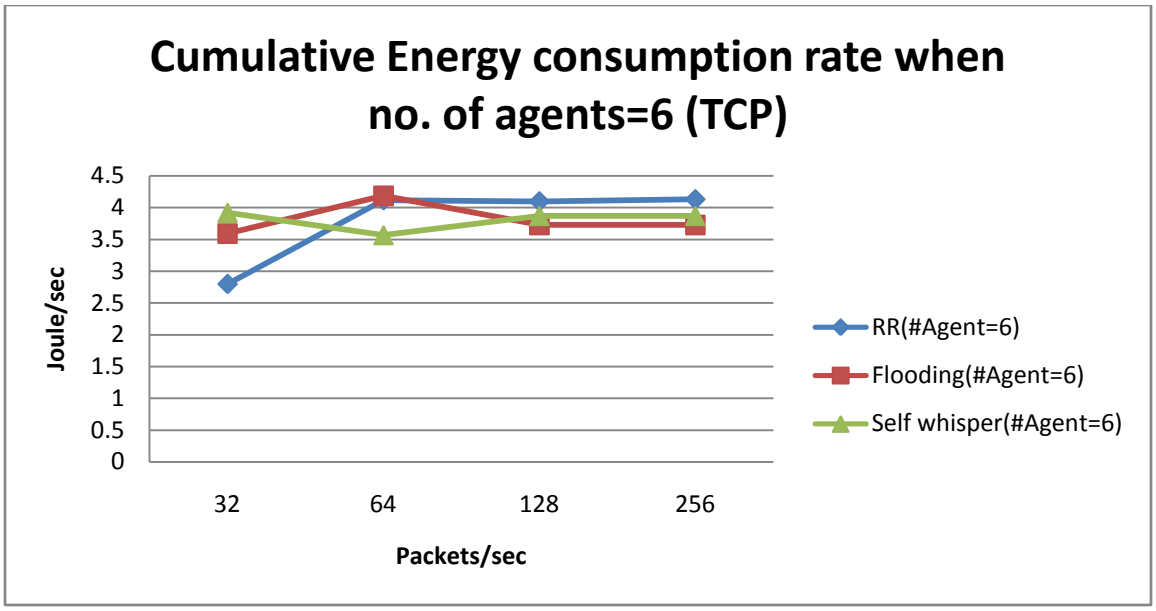


Fig 4.34 Cumulative energy consumption rate when no. of agents=2

From Fig 4.36, 4.37 and 4.38, it is clear that Round Robin results in lower cumulative energy consumption rate. Hence, if the goal of counter attack is to have agent nodes with the lowest cumulative energy consumption rate, Round Robin may be a best choice.

2. Comparison based cumulative packet drop rate for agents

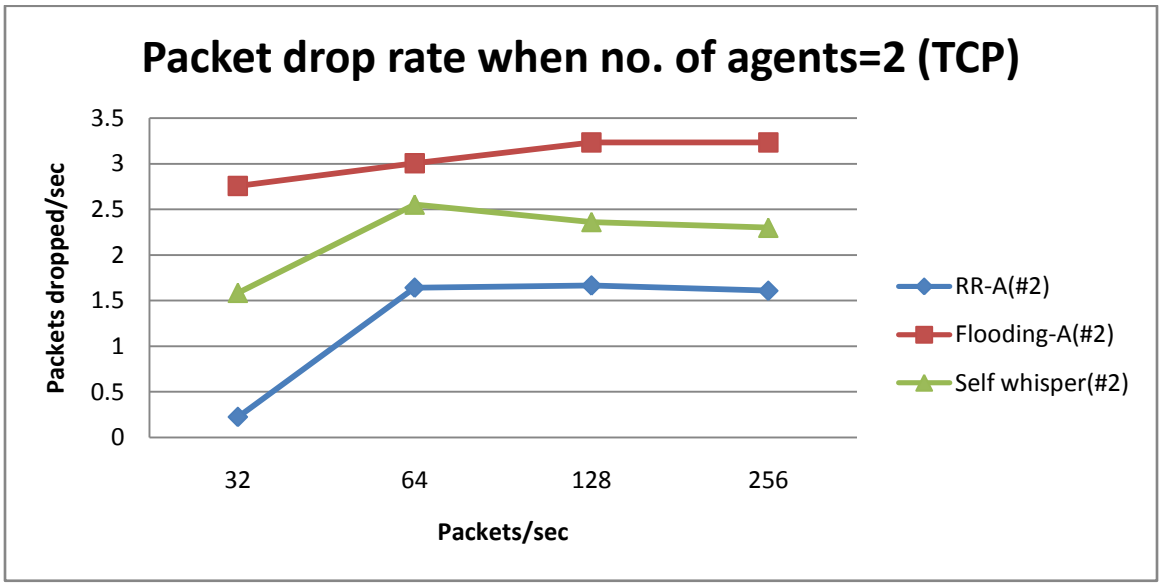


Fig 4.35 Packet drop rate of Agents when no. of agents=2

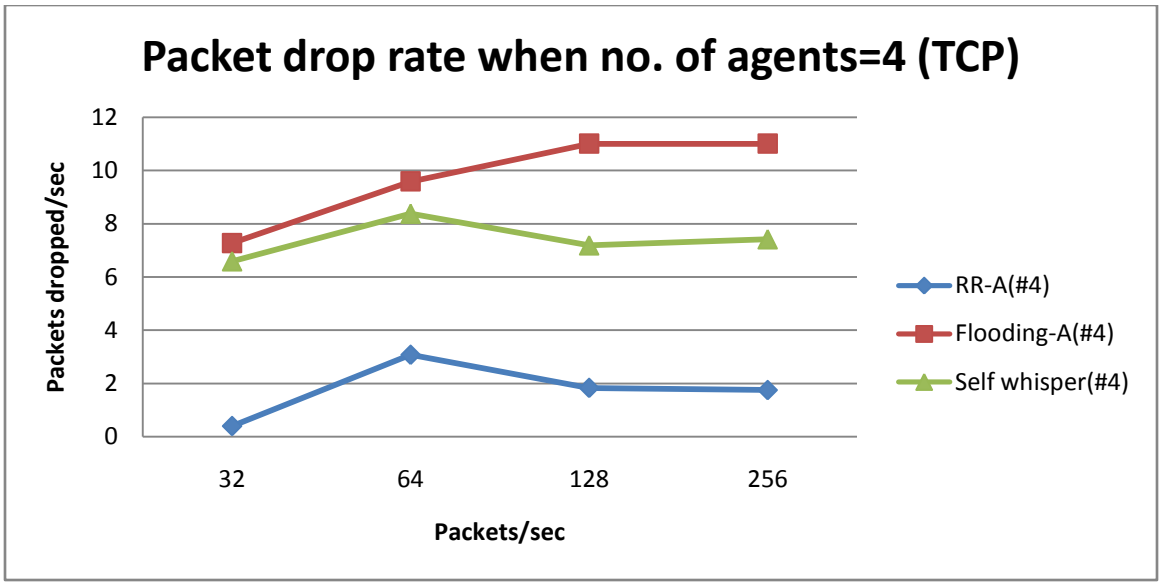


Fig 4.36 Packet drop rate of Agents when no. of agents=4

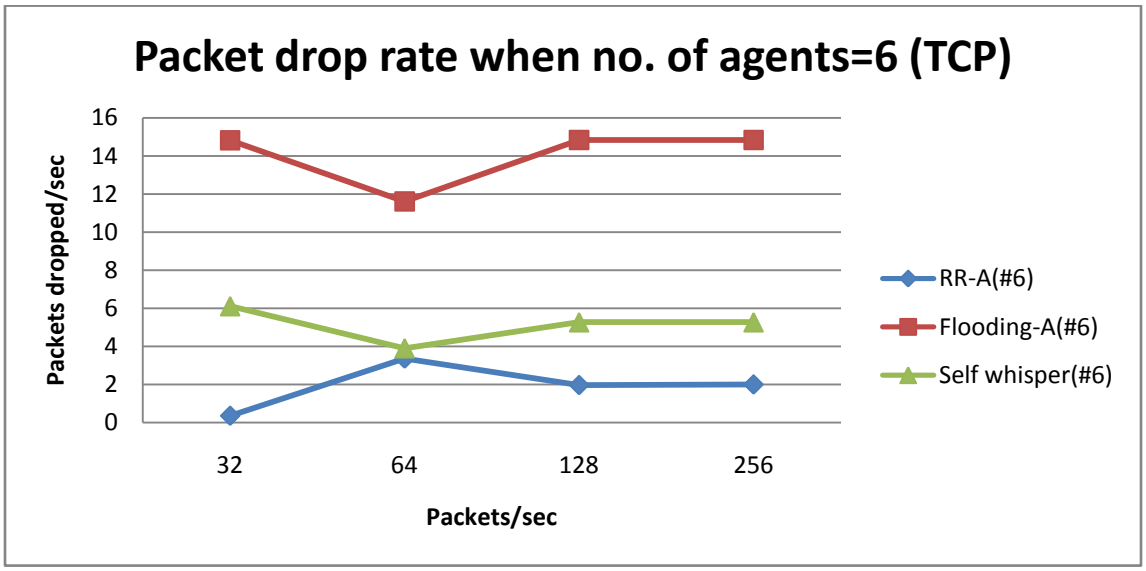


Fig 4.37 Packet drop rate of Agents when no. of agents=6

As we can see, again Round Robin performs better than other two when it comes to the lowest packet drop rate by agent nodes.

3. Comparison based on Lowest time taken to exhaust intruder energy

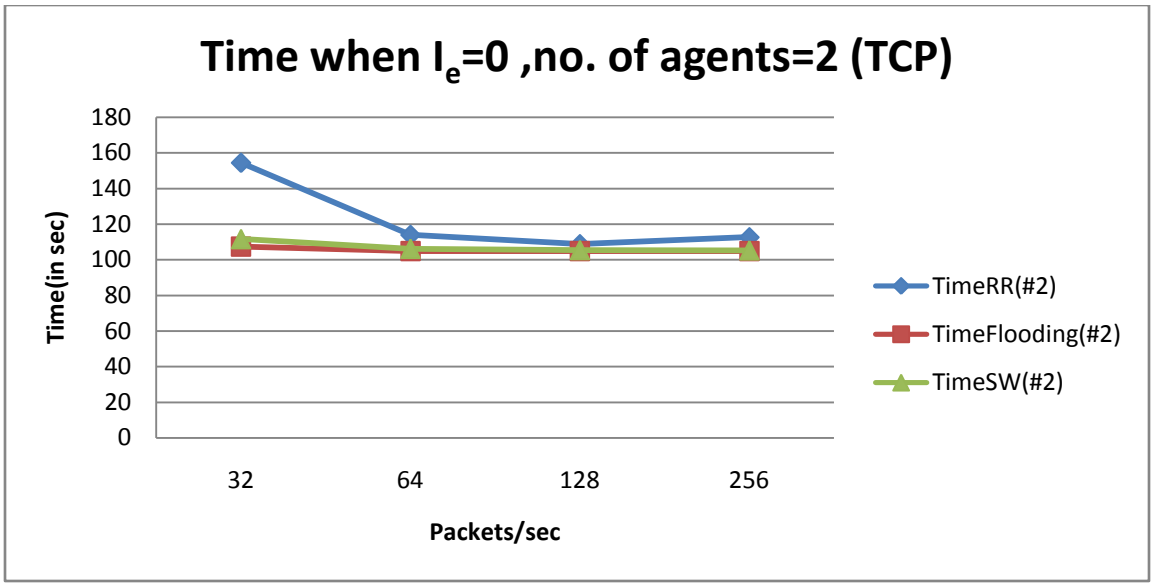


Fig 4.38 Time when I_e=0, no. of agents=2, TCP

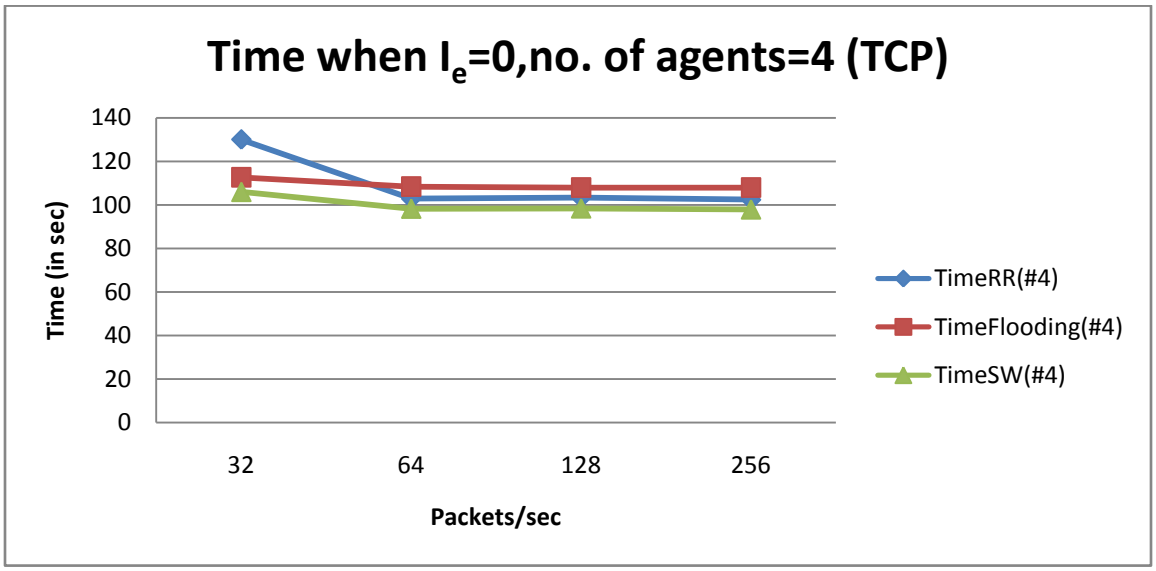


Fig 4.39 Time when I_e =0, no. of Agents=4, TCP

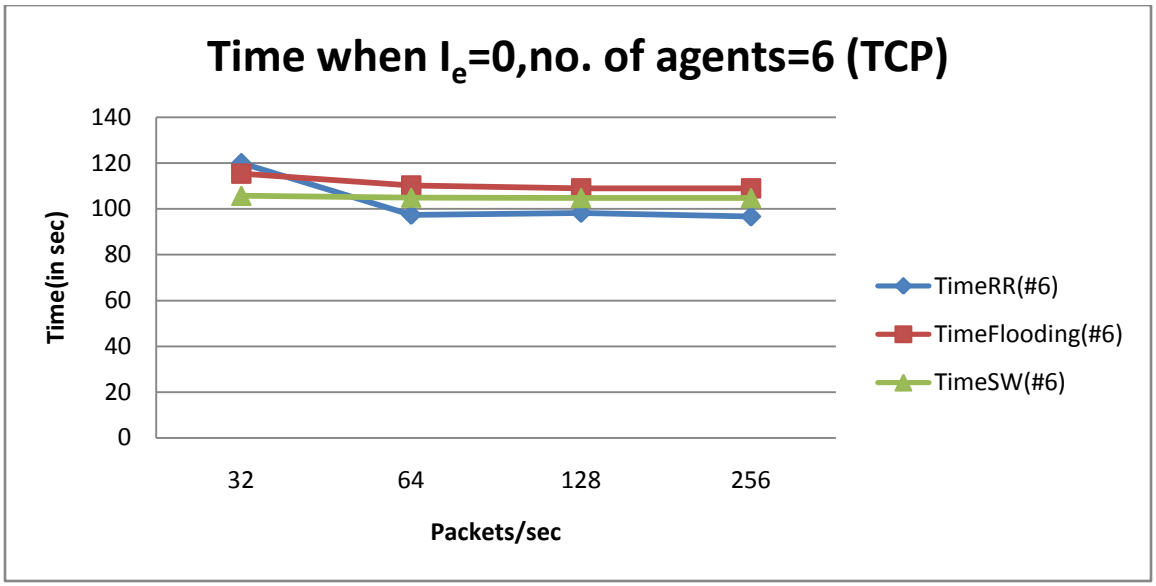


Fig 4.40 Time when I_e =0, no. of Agents=6, TCP

It is clear from above the comparison that, self whisper yields the lowest time required to exhaust intruder energy. Based on three parameters mentioned at beginning of this section, Round robin performs better than the other two of models for first two parameters. However, for 3rd parameter, self whisper has better performance then Round Robin.

Now, we compare results of the three models from a point of view where impact of attack on intruder is maximum. The parameters are:

1. Highest energy consumption rate
2. Highest packet drop rate
3. Least time required to consume entire energy

1. **Comparison to find model that results in higher energy consumption rate for Intruder**

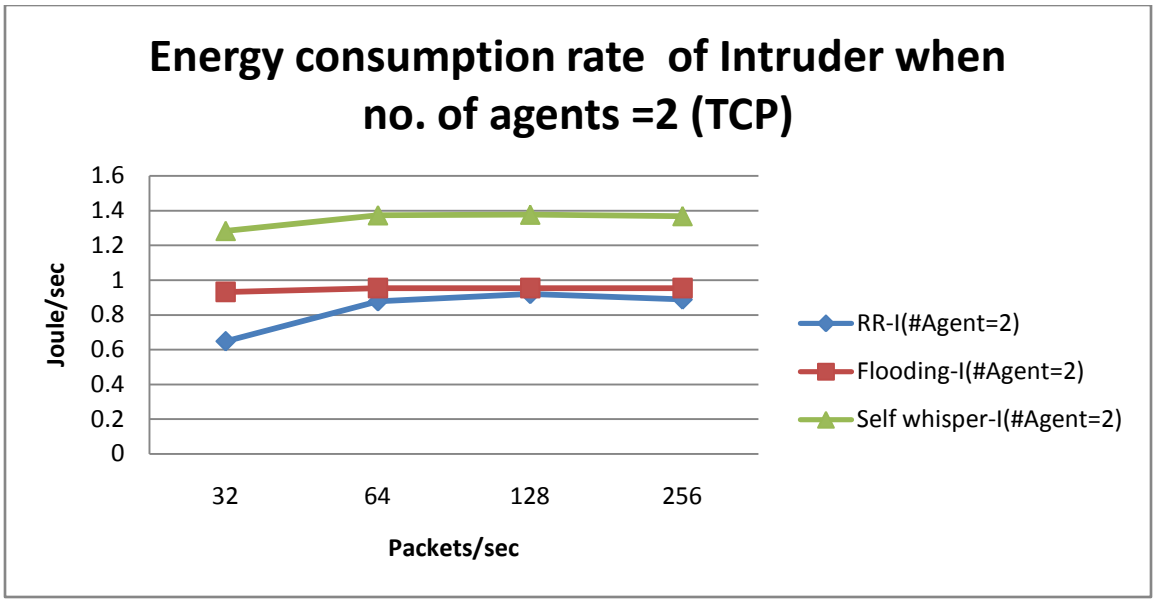


Fig 4.41 Energy consumption rate of Intruder when no. of Agents=2

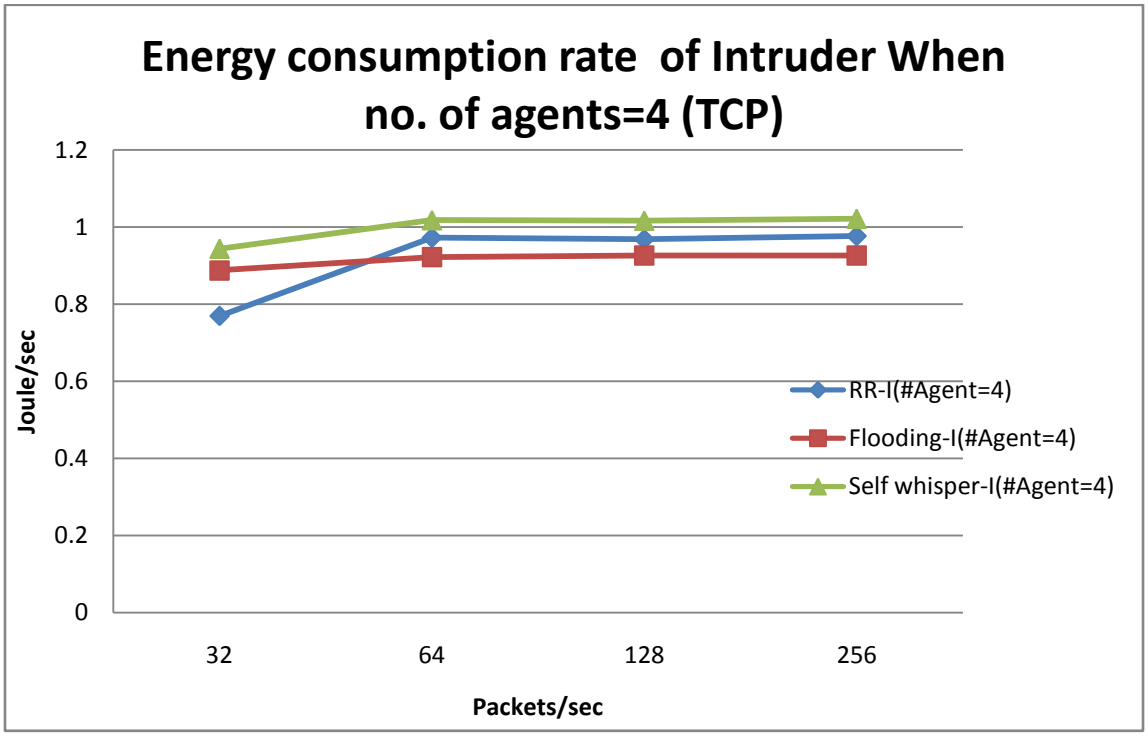


Fig 4.42 Energy consumption rate of Intruder when no. of Agents=4

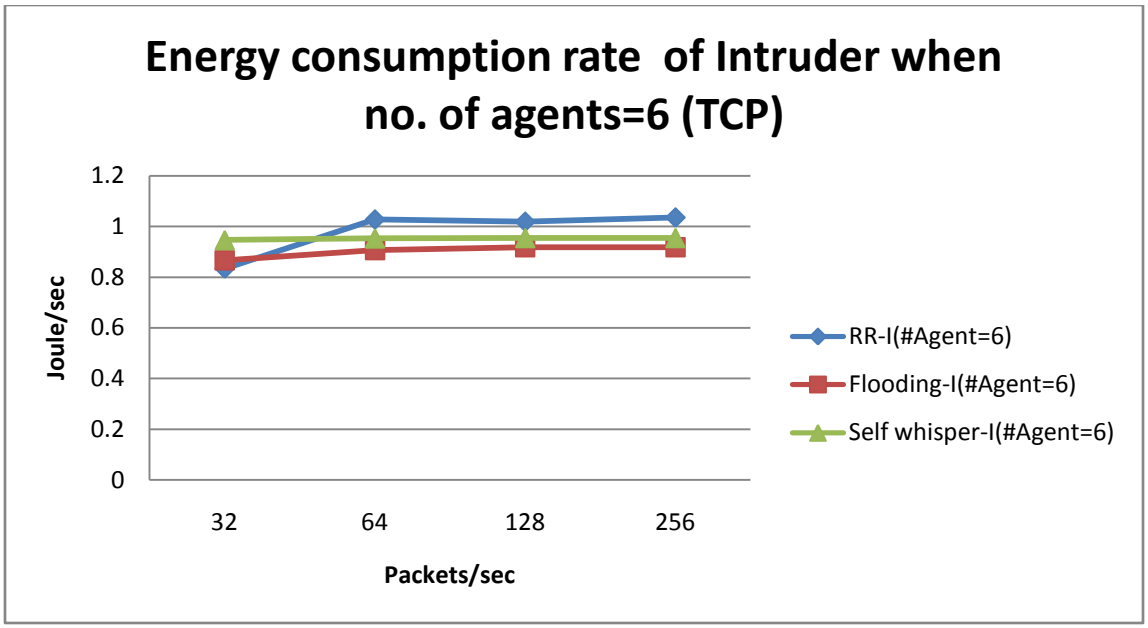


Fig 4.43 Energy consumption rate of Intruder when of Agents=6

Self whisper yields the best result if the objective is to have higher energy consumption rate for Intruder.

2. **Comparison to find best model that results in higher packet drop rate**

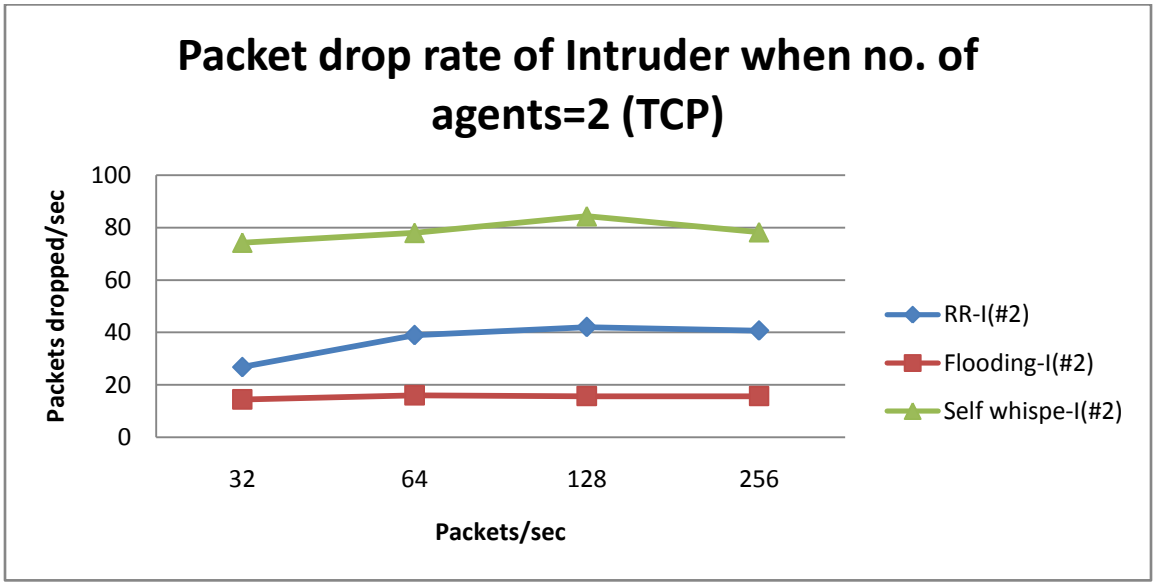


Fig 4.44 Packet drop rate of Intruder when no. of Agents=2

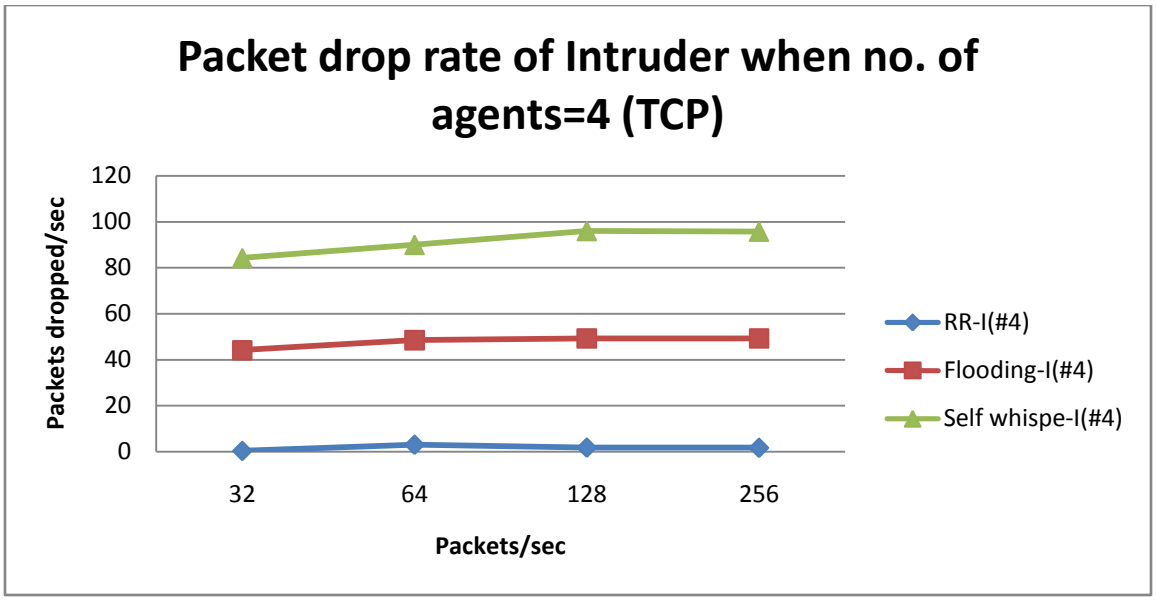


Fig 4.45 Packet drop rate of Intruder when no. of Agents=4

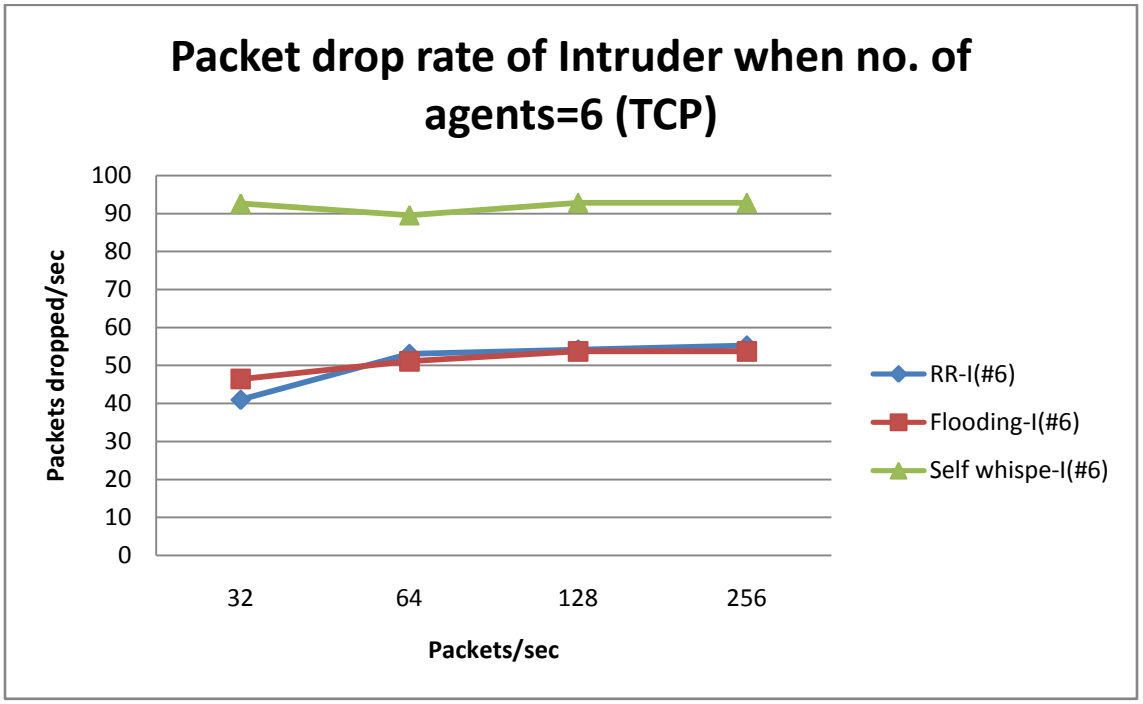


Fig 4.46 Packet drop rate of Intruder when no. of Agents=6

Clearly, the counter-attack model that results in higher packet drop rate for Intruder is Self whisper.

3. **Comparison to find best model that results in least time needed to exhaust intruder energy**

As we saw earlier, the self whisper model is the best choice to exhaust intruder energy in the least time. Hence, considering all three parameters, self whisper is the best model to counter attack an intruder.

For Agents		For Intruder	
Parameter	**Best Model**	**Parameter**	**Best Model**
Min cumulative energy consumption rate	Round Robin	Max energy consumption rate	Self Whisper
Min cumulative packet drop rate	Round Robin	Max packet drop rate	Self Whisper
Min Time needed to exhaust intruder energy	Self Whisper	Min Time needed to exhaust intruder energy	Self Whisper

Table 4.6 Model comparison for TCP version

For UDP:

As for TCP, the best scheme for UDP was studied the findings are summarized below.

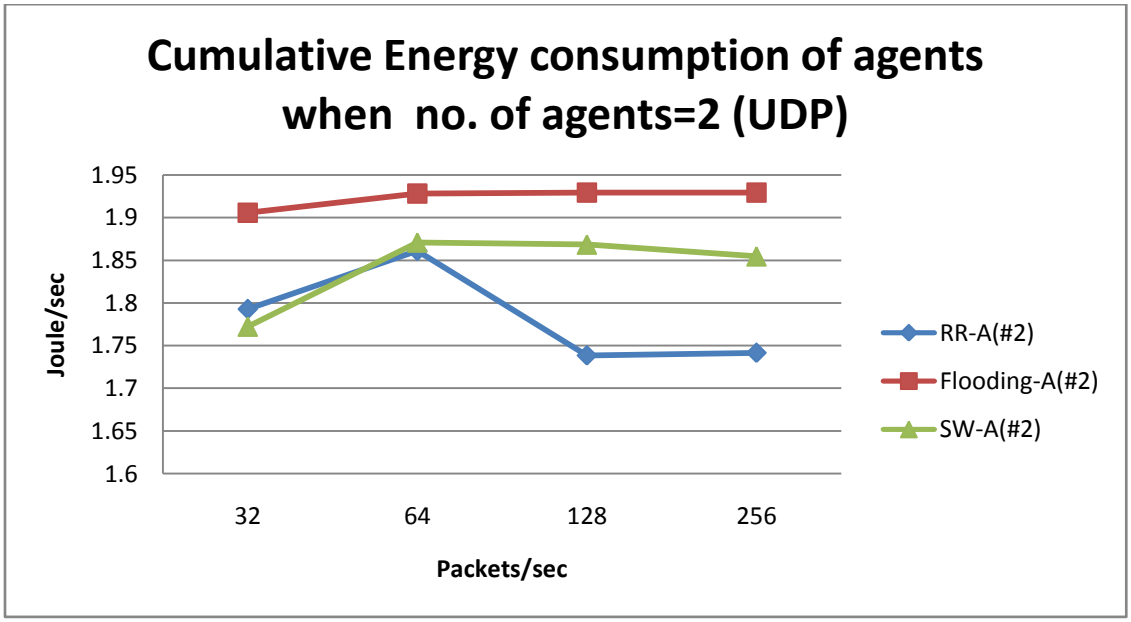


Fig 4.47 Cumulative Energy consumption rate of Agents when no. of Agents=2

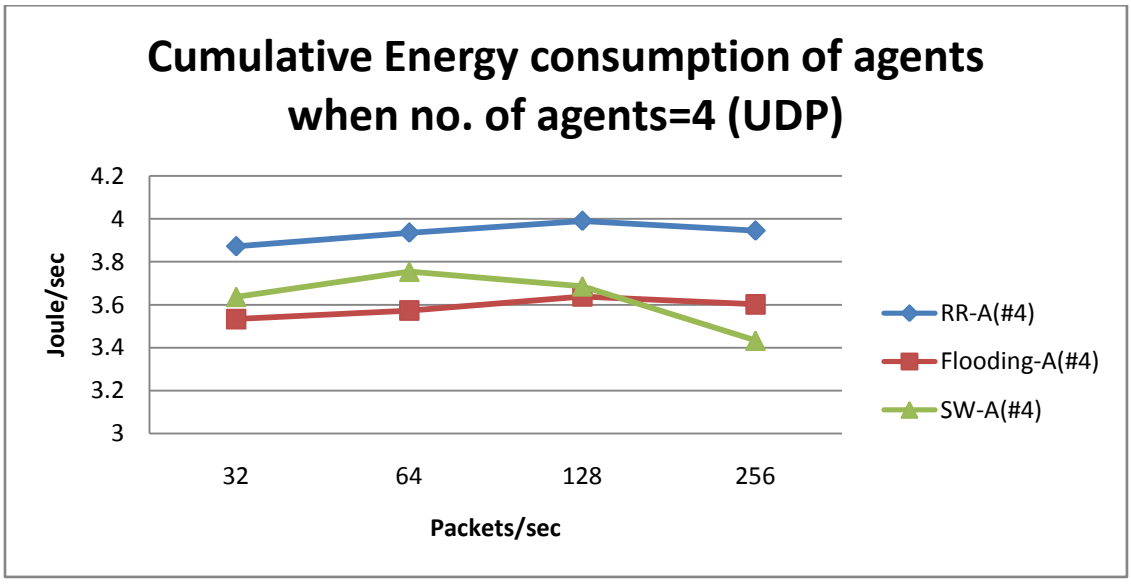


Fig 4.48 Cumulative Energy consumption rate of Agents when no. of Agents=4

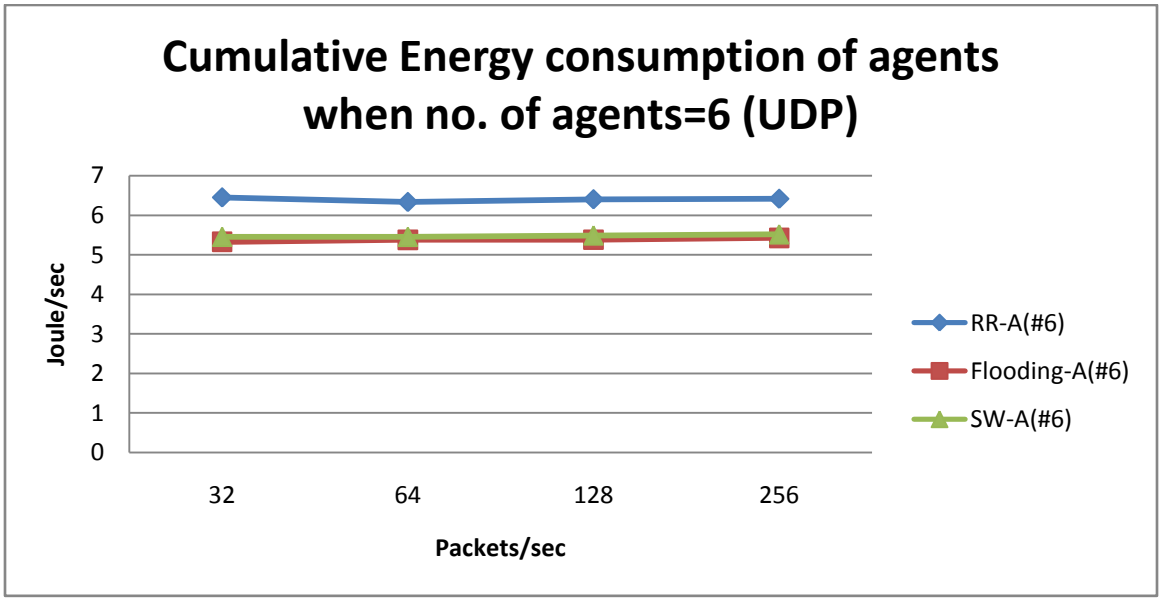


Fig 4.49 Cumulative Energy consumption rate of Agents when no. of Agents=6

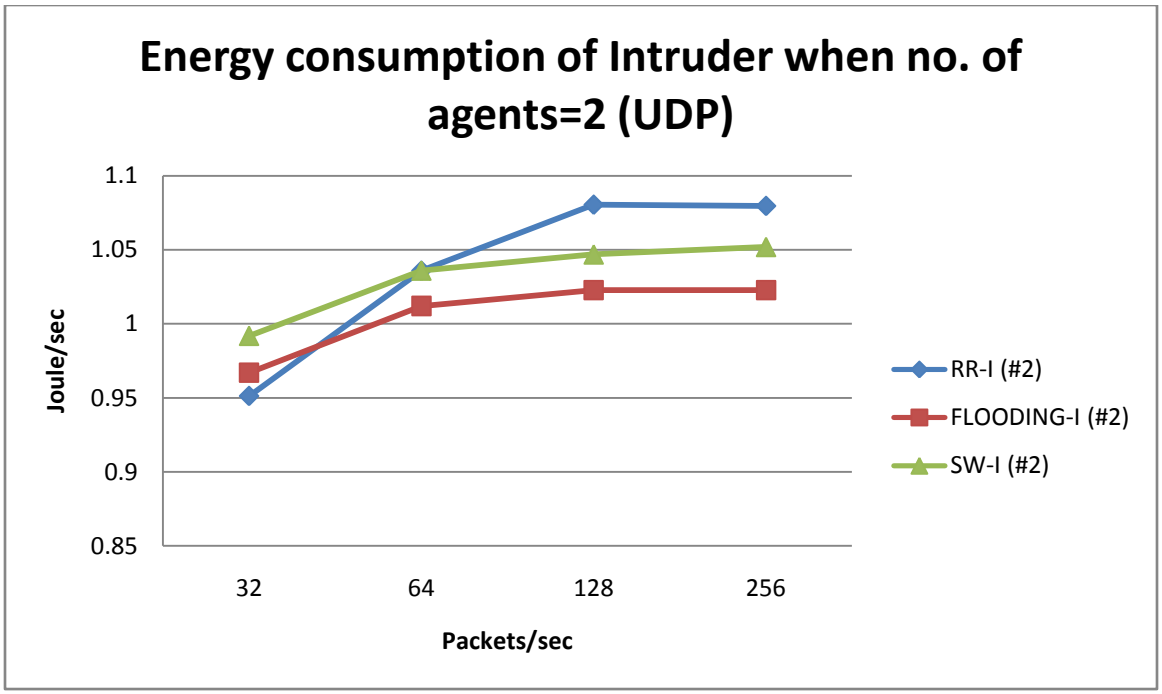


Fig 4.50 Energy consumption rate of Intruder when no. of Agents =2

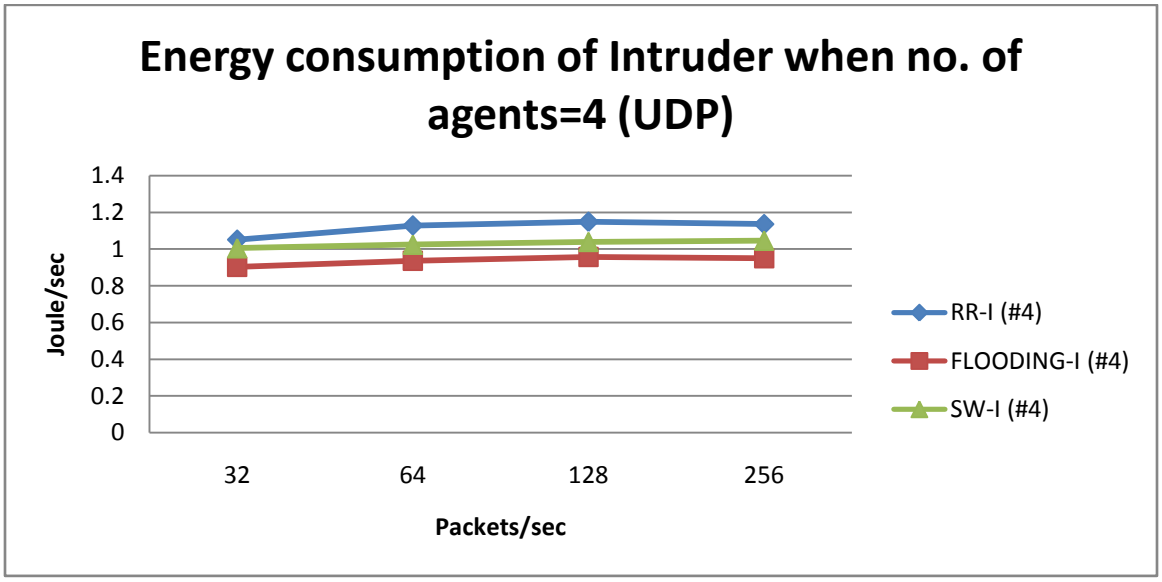


Fig 4.51 Energy consumption rate of Intruder when no. of Agents =4

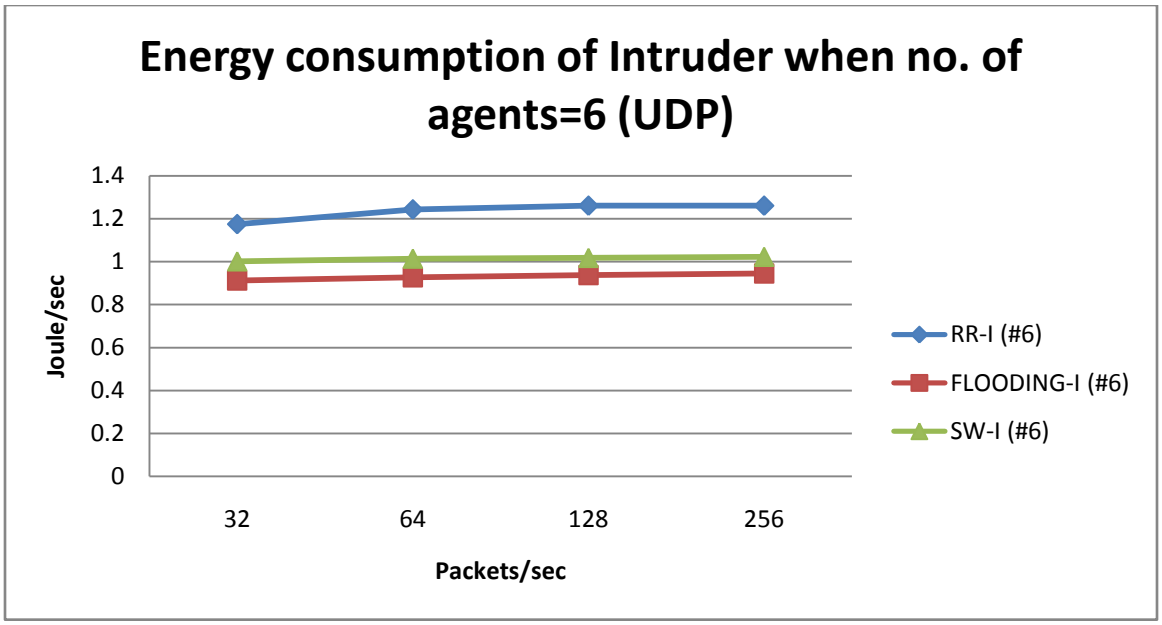


Fig 4.52 Energy consumption rate of Intruder when no. of Agents =6

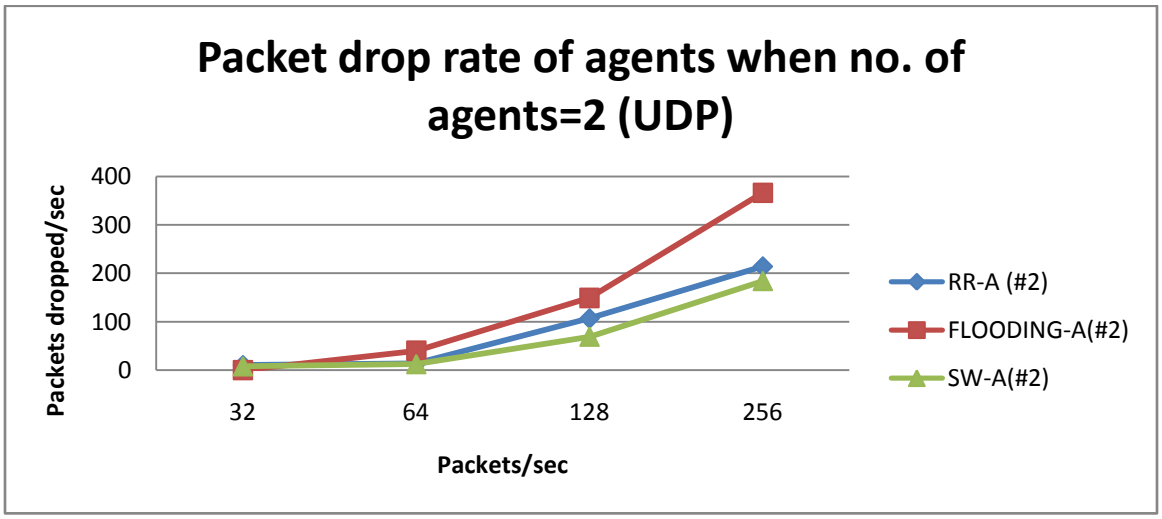


Fig 4.53 Cumulative Packet drop rate of Agents when no. of Agents =2

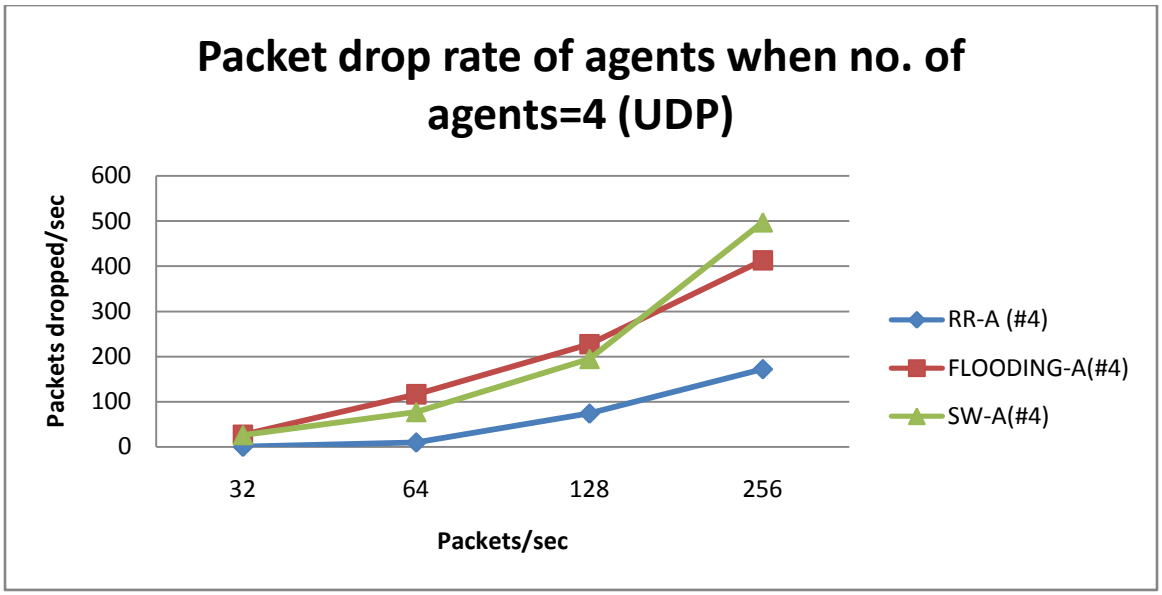


Fig 4.54 Cumulative Packet drop rate of Agents when no. of Agents =4

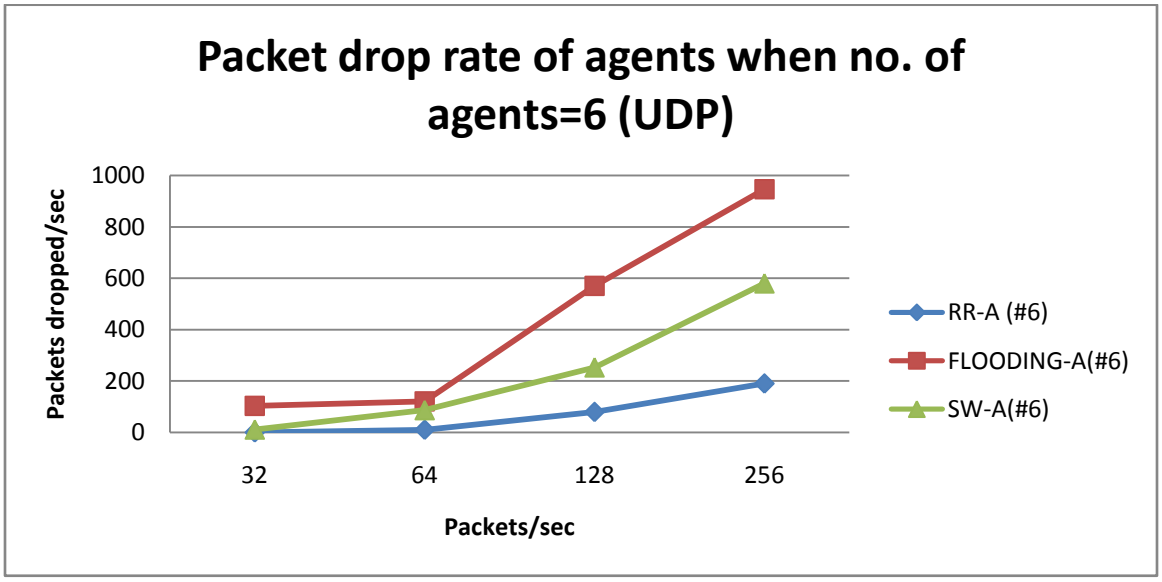


Fig 4.55 Cumulative Packet drop rate of Agents when no. of Agents =6

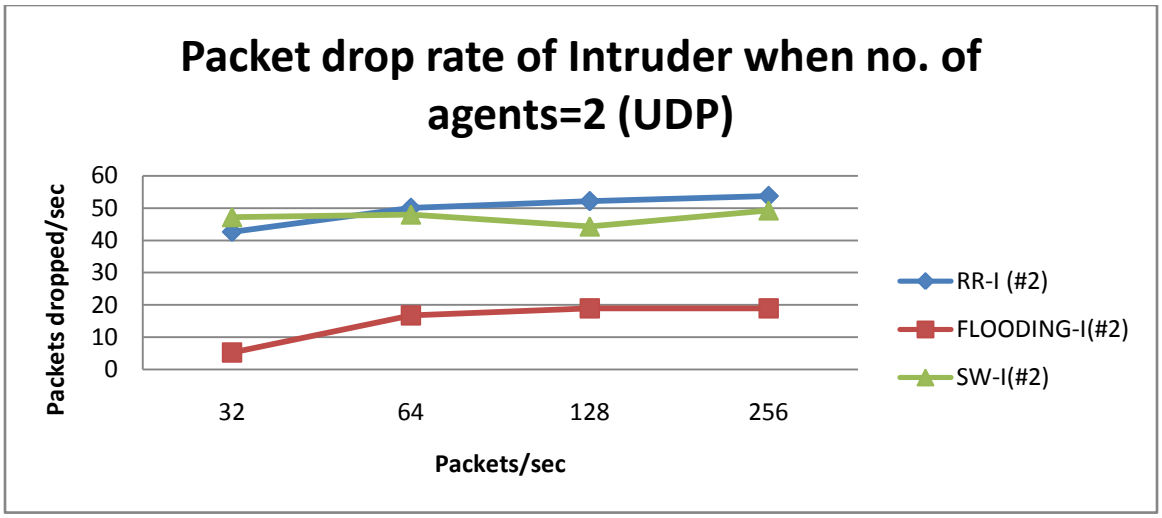


Fig 4.56 Packet drop rate of Intruder when no. of Agents =2

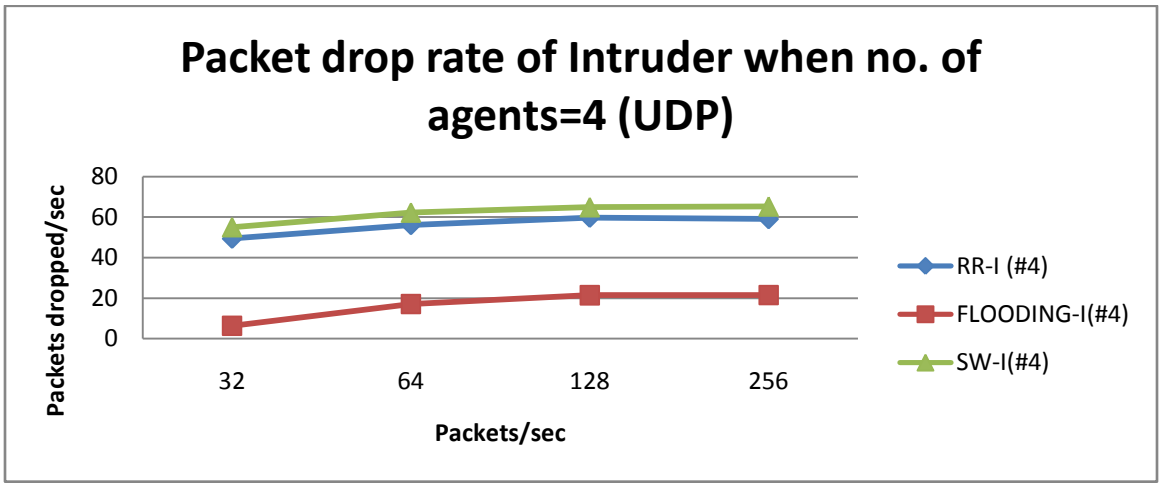


Fig 4.57 Packet drop rate of Intruder when no. of Agents =4

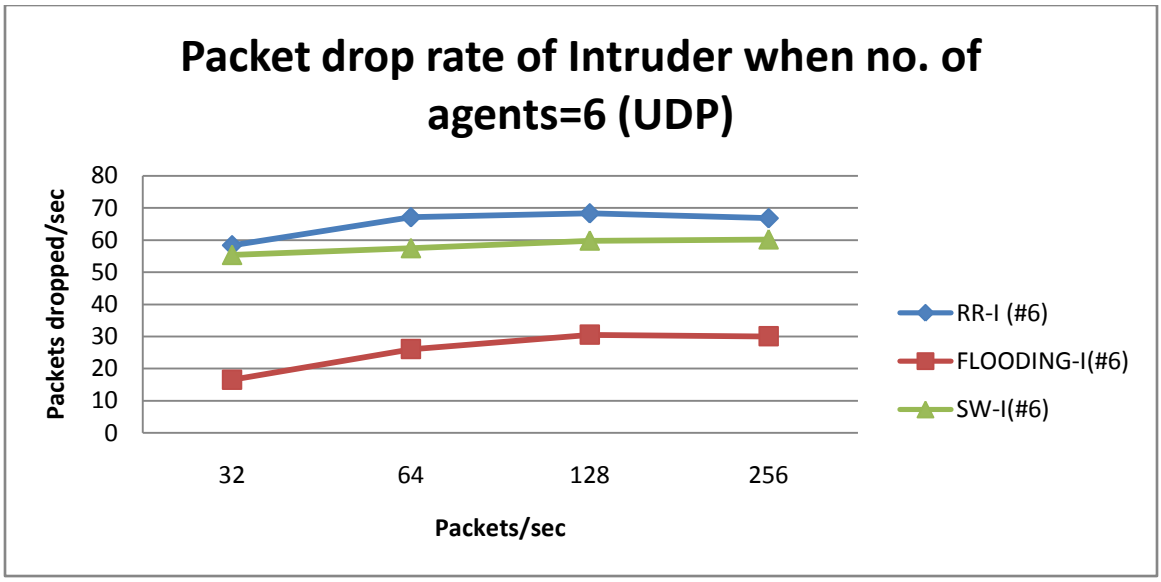


Fig 4.58 Packet drop rate of Intruder when no. of Agents =6

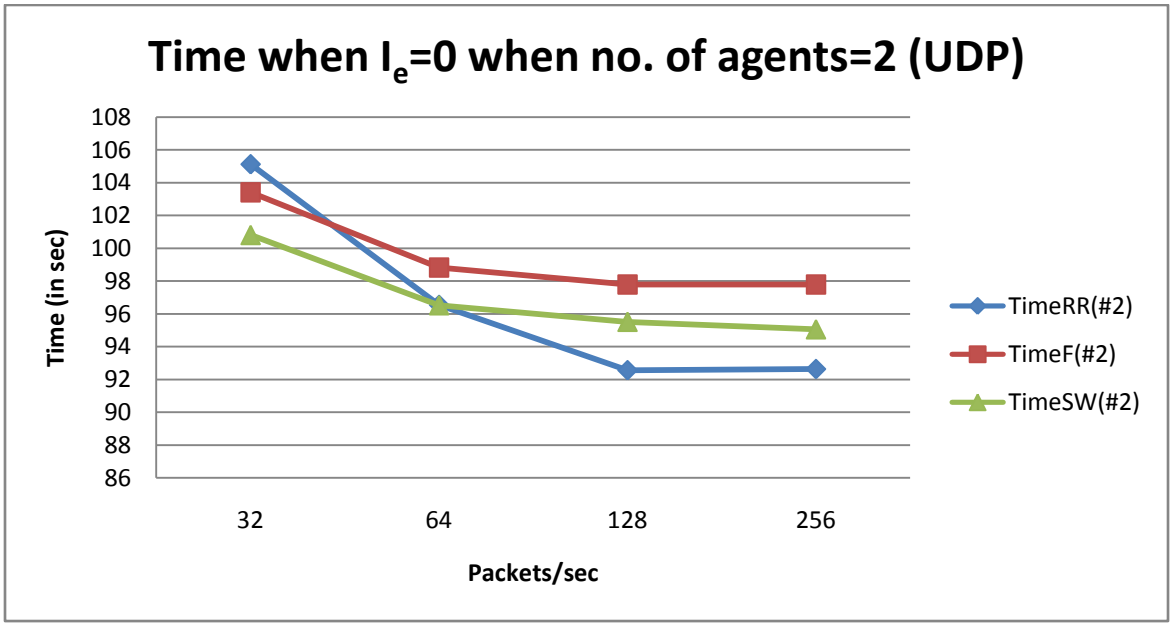


Fig 4.59 Time taken to exhaust intruder energy when no. of Agents =2

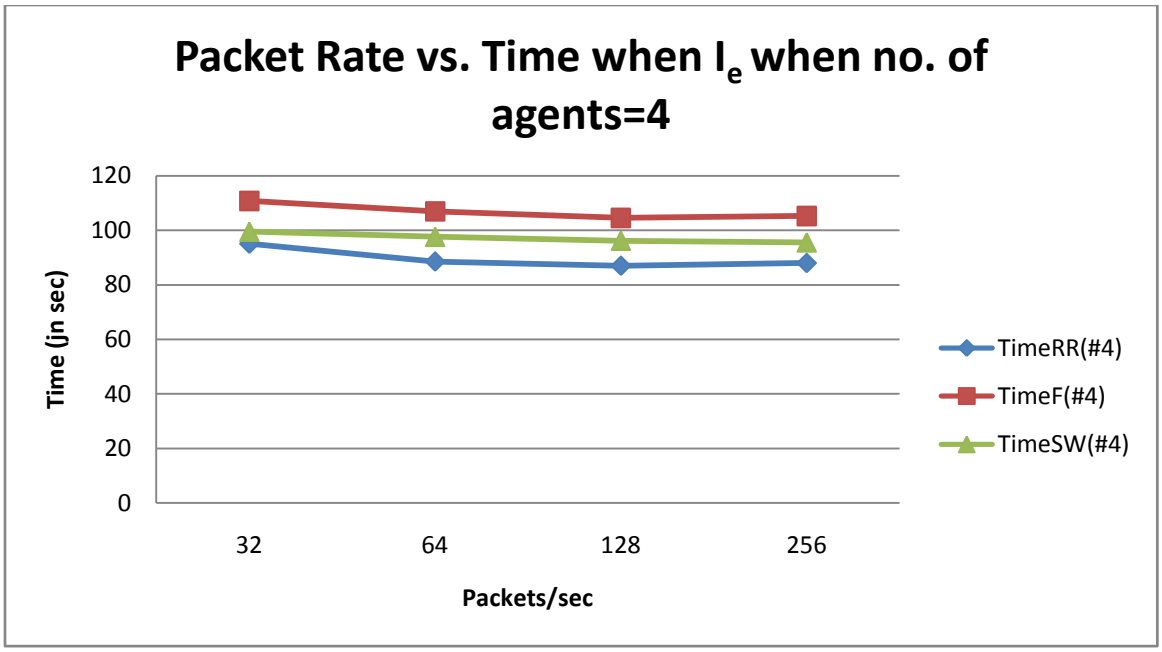


Fig 4.60 Time taken to exhaust intruder energy when no. of Agents =4

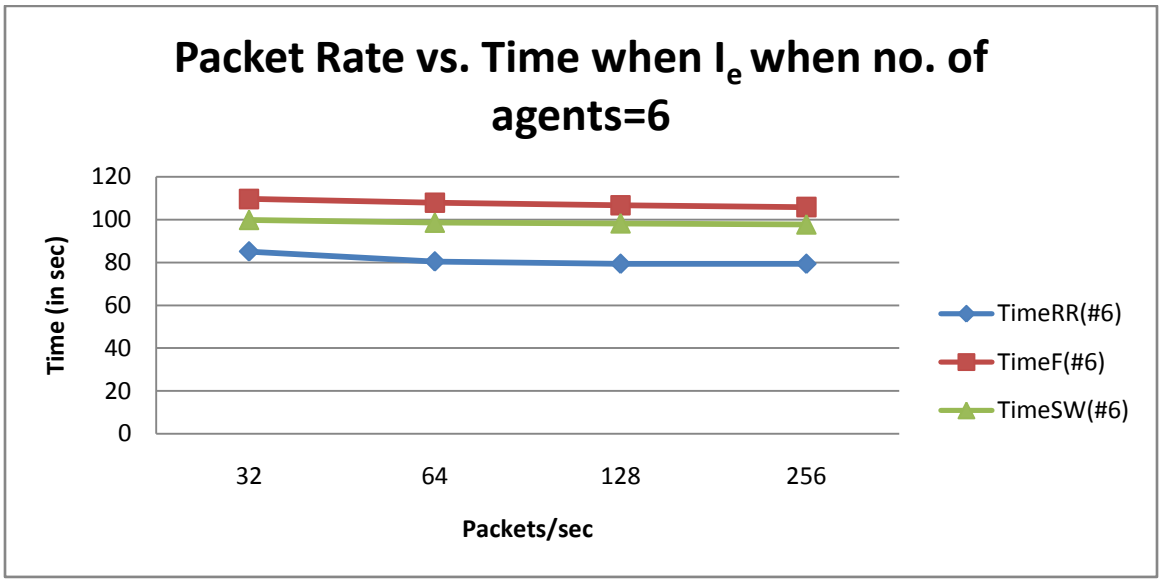


Fig 4.61 Time taken to exhaust intruder energy when no. of Agents =6

For Agents		For Intruder	
Parameter	**Best Model**	**Parameter**	**Best Model**
Min cumulative energy consumption rate	Flooding	Max energy consumption rate	Flooding

Min cumulative packet drop rate	Round Robin	Max packet drop rate	Round Robin
Min Time needed to exhaust intruder energy	Round Robin	Min Time needed to exhaust intruder energy	Round Robins

Table 4.7 Model comparison for UDP version

If the protocol used during counter attack is UDP, then as is evident from the table above, Round Robin is the best model to use. It results in minimum packet drop rate by agent nodes and also takes minimum time to overpower the intruder.

CHAPTER V

CONCLUSIONS

The thesis proposes new way of responding to an intruder in an ad hoc wireless network. Taking an offensive approach against an intruder has some basic advantages like reduced rate of successful attack, extra layer of security, wasting intruder's resources etc. The choice of counter-attack model depends upon the objective of the counter-attack. If objective is to exhaust intruder's critical resource like energy, bandwidth in the least amount of time, and if the protocol used is TCP, self whisper is definitely a best choice. But if the objective is to have minimum energy consumption rate and minimum packet drop rate for agent nodes, Round Robin is the best choice with TCP.

On the other hand, if the protocol used for counter-attack is UDP, and the objective is to consume maximum amount of intruder energy, self whisper performs better but if the objective is to have maximum packet drop rate at the intruder in the minimum time period, Round Robin is perhaps a best choice.

The optimal packet transmission rate for the proposed models is <= 64. Beyond this limit, an increase in packet transmission rate has little or no effect on the outcome. Another important point to mention is that, simply increasing the number of agent nodes does not yield a better result. Instead, smaller group of agent nodes appear to perform better then a large one.

We believe counter-attack is a promising approach in many special applications such as a the battlefield. This is especially important if the attacker is more intelligent and powerful in terms of resources such as energy, communications, processing power and buffer size. Therefore, future research will focus on exploiting the scenarios where an intruder is more powerful than an individual agent node. A scenario where multiple intruders are attacking the network and network administrator has limited number of agents is another area for future work. This thesis has focused on exploiting the idea of counter-attacks. There are many questions which remain unanswered in this thesis; these include:

1. How the models will perform if the intruder is more powerful than ordinary nodes.
2. How to respond if multiple intruders are executing multiple type of attacks in the network?
3. In the case of multiple intruders launching multiple type of attacks, how is a network manager to decide which attack to respond to first? What should be the parameters based on which a network manager can take a decision?
4. How to prioritize among several attacks launched by Intruders?
5. What are the legal consequences if agent nodes attack an ordinary node due to mistaken identity?

These are questions that need to be addressed in future work.

REFERENCES

[1] Mohit Jain and Himanshu Kandwal, "A survey on complex wormhole attack in wireless ad hoc networks" *Proceedings 2009 International Conference on Advances in Computing, Control and Telecommunication Technologies*, pp. 555-558, 2009

[2] James Newsome, Elaine Shi, Dawn Song and Adrian Perrig "The Sybil attack in sensor networks: Analysis and Defenses", *Proceeding* 2004 *IEEE/ACM International Conference on Information Processing in Sensor Networks ACM* April 26-27, pp. 20-28, 2004

[3] N.Bhalaji, Dr. A.Shanmugam "Association between nodes to combat black hole attack in DSR based MANET", *Proceeding 2009 IEEE Infocom,* pp. 200-203, 2009

[4] C. Perkins and E Royer, "Ad Hoc On-Demand Distance Vector Routing", *Proceeding 1999 IEEE Workshop on Mobile computing systems and Applications,* pp. 90-100, 1999

[5] Yih-Chun Hu, Adrian Perrig, David B. Johnson, "Packet Leaches: A defense against wormhole attacks in wireless ad-hoc network", *Proceeding 2003 22nd Annual Joint Conference of the IEEE Computer and Communications Sciences (INFOCOM*), pp.1976-1986, 2003

[6] S.Marti, T.J.Guili,K.Lai,M.Baker, "Mitigating routing misbehavior in mobile ad hoc networks," *Proceeding 2000 6th Annual International Conference on Mobile computing and Networking,* Aug 6-11,Boston,Massachusetts, pp. 255-265, 2000

[7] Animesh Patcha, Amitabh Mishra, "Collebrative security architecture for black hole attack prevention in Mobile ad hoc network" , *Proceeding 2003 Radio and Wireless Conference (RAWCON) IEEE*, pp. 75-78, 2003

[8] J.R. Douceur."The Sybil attack", *Proceeding Mar 2002 First international workshop on peer-to-peer systems (IPTPS '02)*, pp. 200-206, 2002

[9] C.Karlof and D.Wagner, "Secure routing in wireless sensor networks : attacks and countermeasures", *Proceeding May 2003 in First IEEE international workshop on sensor network protocols and applications*, pp. 113-127, May 2003.

[10] Ping Yi, Zhoulin Dai, Shiyong Zhang, Yiping Zhong, "A new routing attack in mobile ad hoc network*", International Journal of Information Technology*, Vol 11, No.2, pp. 83-94, 2005

[11] The Network Simulator – ns- 2, http://www.isi.edu/nsnam/ns/index.html, April 13, 2011

[12] C.K.Toh, *Ad Hoc Mobile Wireless Network-Protocols and systems*, Prentice Hall, 2001

[13] Tutorial for Network Simulator "ns", http://www.isi.edu/nsnam/ns/tutorial/ , April 16, 2011

[14] L. M. Feeney, Martin Nilsson, "Investigating the Energy Consumption of a wireless network Interface in ad hoc wireless networking", *Proceeding April 2001 20th Annual Joint Conference of the IEEE Computer and Communications Sciences (INFOCOM)*, pp. 1548-1557, 2001

[15] Geraud Allard1, Pascale Minet1, Dang-Quan Nguyen1, and Nirisha Shrestha. "Evaluation of the Energy Consumption in MANET", *Proceeding 2006 ADHOC-NOW*, pp.170-183, 2006

[16] "The *ns* Manual (formerly *ns* Notes and Documentation)1", The VINT project A Collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC.

Kevin Fall, Editor, Kannan Varadhan, Editor May 9, 2010, http://www.isi.edu/nsnam/ns/doc/index.html, April 18, 2011

[17] Eitan Altman and Tania Jimenez,"NS simulation for beginners", Univ. de Los Andes,Merida, Venezuela and ESSI, Lecture Notes, 2003-2004, December 4, 2003

[18] Transmission Control Protocol, http://en.wikipedia.org/wiki/Transmission_Control_Protocol , April 18, 2011

[19] User Datagram Protocol, http://en.wikipedia.org/wiki/User_Datagram_Protocol, April 18, 2011

VITA

BIPUL CHANDRA

Candidate for the Degree of

Master of Science

Thesis: COUNTER ATTACK AS DEFENSE MECHANISM IN AD HOC MOBILE WORELESS NETWORK

Major Field: COMPUTER SCIENCE

Biographical:

Education:

Completed the requirements for the Master of Science in Computer Science at Oklahoma State University, Stillwater, Oklahoma in July, 2011.

Completed the requirements for the Bachelor of Science in Information Science at B V Bhoomaraddi College of Engineering and Technology, Hubli, Karnataka, India in 2006.

Experience:

December 2009 – May 2011, Research Assistant at Biosystems and Ag. Engineering Oklahoma State University, Stillwater, OK

Aug 2006 – July 2009, Assistant System Engineer, Tata Consultancy Services Limited, Mumbai, India.

Name: BIPUL CHANDRA Date of Degree: July, 2011

Institution: Oklahoma State University Location: Stillwater, Oklahoma

Title of Study: COUNTER ATTACK AS A DEFENSE MECHANISM IN AD HOC MOBILE WIRELESS NETWORK

Pages in Study: 82 Candidate for the Degree of Master of Science

Major Field: Computer Science

Scope and Method of Study:

Ad-Hoc Mobile Wireless Network (MANETs) have emerged and evolved in many forms. MANETs are rapidly gaining popularity because they do not rely on a pre-infrastructure and can be deployed spontaneously. However, compared to wired networks, MANETs are more vulnerable to security attacks due to their unique features, such as stringent power consumption, error prone communication media and highly dynamic network topology. Most of the work done for improving security are focused on defensive mechanism like firewalls, gateways etc. Little research has been done on more offensive mechanisms to provide security. We propose three counter attack models, namely, Round Robin attack, Self-Whisper and flooding. The goal of all these attacks is to use up the intruder's critical resources like energy, communications bandwidth and force the intruder to eventually enter into DoS status.

Findings and Conclusions:

Simulation results shows that proposed counter attack models are an effective tool to counterattack. Simulation shows that a single model may not perform well in all situations. The choice of counter attack model is highly governed by the objective of counterattack. The Self-Whisper attack is the best if the objective is to have minimum energy consumption rate and minimum packet drop rate for agent nodes. On the other hand, if the protocol used is UDP, and the objective is to consume maximum amount of intruder energy, self- whisper performs better but if the objective is to have maximum packet drop rate at the intruder in the minimum time period, Round Robin is perhaps a best choice. Simulation shows that once counter attack begins, any traffic that is through intruder is disrupted. This disruption causes the ordinary nodes that have been tricked by the intruder advertizing false route information, to seek an alternate path to the destination. Hence, counter attack helps in improving security of the network as a whole.

ADVISER'S APPROVAL: Dr. Johnson P. Thomas