**AN ADAPTIVE MODEL FOR DETECTING DDOS ATTACK ON IPV4 AND IPV6**

BY

**ALABI SAMUEL TOWOJU**

**CSC/13/1353**

**BEING A PROJECT REPORT IN THE DEPARTMENT OF COMPUTER SCIENCE SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE, FEDERAL UNIVERSITY, OYE-EKITI**

**DEDICATION**

This research project is dedicated to almighty God, the maker of the universe, the alpha and omega, the giver of knowledge, the beginning and the end, the lord which is, which was, and which is to come, the almighty, for being my strength, my guide, my shield and also my help throughout my course of study and also to my beloved family who gave me their best in all areas of life.

Office Stamp

**DECLARATION OF ORIGINALITY**

This project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet, etc.) has been acknowledged within the main report to an entry in the References list.

I agree that an electronic copy or hardcopy of this report may be stored and used for the purposes of plagiarism prevention and detection.

I understand that cheating and plagiarism constitute a breach of University Regulations and will be dealt with accordingly.

**Copyright**

The copyright of this project and report belongs to Federal University, Oye-Ekiti.

Signed: Date:

**CERTIFICATION**

This project report with the title

**AN ADAPTIVE MODEL FOR DETECTING DDOS ATTACK**

**ON IPV4 AND IPV6**

Submitted by

**ALABI SAMUEL TOWOJU**

CSC/13/1353

Has satisfied the regulations governing the award of the degree of

**Bachelor of Science (B.Sc.) in Computer Science**

Federal University, Oye Ekiti, Ekiti State, Nigeria.

…………………………….. ………………………………

Dr. G. OGUNLEYE Date

Supervisor

……………............................... ……………………………………

Dr. OBE Date

Head of Department

**ABSTRACT**

As the Internet is growing so is the vulnerability of the network. Denial of Service attacks (DDoS) are one of such kind of attacks. In this paper, one of the more popular DDoS attack is the TCP-SYN Flood attack. The SYN flooding attacks are launched by exploiting the TCP’s three-way handshake mechanism and its limitation in maintaining its half-opened connections on internet protocols IPv4 and IPv6.

This study is aimed in the detection of DDOS attack with neuro-fuzzy algorithm combination of fuzzy logic and neural network (ANFIS). To simulate this project research MATLAB 2012a software which is a programming language and environment for scientific computing. The result of comparison showed that the ANFIS model to the ANFIS has more accuracy in detecting DDoS in Internet Protocol (IPv4 and IPv6).

**ACKNOWLEDGEMENTS**

Its somehow funny how time flies, just like yesterday when the journey started, though it looks so challenging and seem to be tough, I bless the name of the lord for his infinite love and his tender mercy towards me, enabling me to write this project and for making this day a reality in my life.

I highly appreciate the effort and guidance of my supervisor, **Dr. G. OGUNLEYE** for taking his time to go through the report and make corrections where necessary. I also acknowledge the support of member of staff of the department. I pray good lord will reward your labour in Jesus name (amen),

My sincere appreciation also goes to my parent **Mr. and Mrs. Alabi** for their ceaseless prayer, advice, financial and moral support and also for their assistance throughout the course of my study. May God almighty bless you and I pray you will eat the fruit of your labour in Jesus name.

**TABLE OF CONTENT**

**CONTENT**  **PAGES**

Cover Page i

Title Page ii

Certification iii

Declaration of Originality iv

Dedication v

Acknowledgments vi

Abstract vii

Table of Contents viii

List of Figures x

**CHAPTER ONE**

INTRODUCTION

1.0 Project Synopsis 1

1.1 Background to the Study 1

1.2 Statement of the Problem 3

1.3 Motivation 4

1.4 Aim and Objectives 5

1.5 Contribution to knowledge 5

1.6 Project Arrangement 5

**CHAPTER TWO**

LITERATURE REVIEW

2.0 Introduction 7

2.1 Historical background of IPv4 and IPv6 11

2.1.1 Ipv6 Improvement Over Ipv4 12

2.1.2 Denial of Service in Ipv6 Network 13

2.1.3 Internet Protocol (Ipv4 And Ipv6) Address Security 13

2.2 Existing methods for DDoS Attack Detection 14

2.2.1 Algorithms and Techniques for Detecting DoS / DDoS Attacks on 16

Network Servers and Internet Protocols

2.3 Review of Adaptive Neuro Fuzzy Inference Scheme (ANFIS) 22

**CHAPTER THREE**

PROJECT METHODOLOGY

3.0 Introduction 25

3.1 Methodology 25

3.2 Requirement specification 25

3.2.1 Functional Requirements 26

3.2.2 Non-Functional Requirements 26

3.2.3 Software Requirements 27

3.2.4 Hardware Requirements 27

3.3 System Analysis 28

3.3.1 Overview of Various DDoS Attack 28

3.3.2 ANFIS (FIS) Structure and Parameter Adjustment 29

3.3.3 A Normal and Attack Scenario 30

3.3.4 Protocol to trace back the source of DDoS attacks 34

using Neuro-Fuzzy Algorithm.

3.4 Method of Data Collection 35

3.4.1 Preprocessing of Datasets 35

3.5 Design 36

3.5.1 Evaluation Metric 36

3.5.2 Design of Proposed Architecture 37

**CHAPTER FOUR**

IMPLEMENTATION, RESULTS AND DISCUSSION

4.0 Introduction 41

4.1 Network Simulation 41

4.2 Testing 42

4.2.1 Training Data 43

4.3 Project schedule 46

4.4 Quality management 47

**CHAPTER FIVE**

CONCLUSION

5.0 Conclusion 48

5.1 Contribution to knowledge 48

5.2 Limitations 49

5.3 Recommendation and future works 49

5.4 Critical Appraisal 49

REFERENCES 50

APPENDIX 53

**LIST OF FIGURES**

**PAGES**

Figure 3.1: DDoS Attack Overview. 29

Figure 3.2: ANFIS Model Structure. 30

Figure 3.3: A Normal Scenario and A (SYN Flood Attack) Scenario. 32

Figure 3.4: A TYPICAL SYN-Flood Attack. 33

Figure 3.5: Basic Flow of Designing Artificial Neural Network Model 38

Figure 3.6: The Activity Flow Diagram of proposed method. 40

Figure 3.7: DDoS Detection Flowchart. 41

Figure 3.8: Proposed Architecture for Network Traffic Analyzer 42

Figure 4.1: Comparison of Training Data and ANFIS data 46

Figure 4.2: ANFIS Training Data Error at each training epoch. 47

Figure 4.3: Root Mean Squared Checking Data Errors at each training epoch. 48

Figure 4.4: Detection (Snapshot of the Interface) 49

**CHAPTER ONE**

**INTRODUCTION**

This chapter focuses on the introductory aspect of the project, it consists of the background of the research project, statement of the problem, project aim and objectives and observation. It also gives an overview of the project report structure.

**1.1 BACKGROUND OF THE STUDY**

A Denial-of-Service (DoS) attack is a network attack from a single machine that attempts to prevent the victim, the targeted machine, from communicating to other devices on the network or perform its normal tasks (DiMarco, 2012). The extension of these attacks to include many malicious machines became known as Distributed Denial-of-Service (DDoS) attacks. DDoS attacks causes an immense amount of strain on both the victim and the devices used to reach the victim (DiMarco, 2012).

According to Manickam, (2014), the first well documented DoS attacks occurred in 1974. These attacks were developed by hackers to disrupt communication between a client and a server. They would be targeted against a victim machine, but can lead to other machines being affected. Depending on the attack, the victim could fail to provide a single service or fail to provide any network connectivity at all.

One of the major challenges in the fast networks security management is that the detection of suspicious anomalies in network traffic patterns is often difficult and the machine will become vulnerable to attacks with time (Redhwan, 2014). A DDoS attack only differs with DoS from the method, a DoS is made from a system or network while a DDoS attack is organized to happen simultaneously from a large number of systems or networks.

A hacker begins a DDoS attack by exploiting vulnerability in a computer system and making it the DDoS "master". From the master system, the intruder identifies and communicates with other systems that can be compromised also. The intruder loads DDoS attack tools on those compromised systems. The intruder can instruct the controlled machines to launch one of many flood attacks against a specified target. The inundation of packets to the target causes a denial of service (Cai and Hembroff, 2006). Some DDoS attacks utilize internet worms to automate the process of exploiting and compromising computer systems, as well as launching DDoS attacks.

Attackers use spoofed source addresses to hide their identity and location in DDoS attacks. Some service providers do perform ingress filtering to check for valid source IP addresses coming into access routers, but this is not completely effective. The trace back mechanisms trace the true source of the attackers to stop the attack at the point nearest to its source to reduce waste of network resources and to find the attacker’s identities (Meena and Trivedi, 2012).

Nowadays, many companies and/or governments require a secure system and/or an accurate Intrusion Detection System (IDS) to defend their network services and the user’s private information. Kato and Klyuev, (2014) research further on network security, and they deduce DDoS attacks jam the network service of the target using multiple bots hijacked by crackers and send numerous packets to the target server.

Servers of many companies and/or governments have been victims of the attacks. In such an attack, detecting the crackers is extremely difficult, because they only send a command by multiple bots from another network and then leave the bots quickly after command execute.

In general, detection is required before the spread of a DDoS attack. DDoS detection is often part of a wider intrusion detection system (IDS). IDS can be classified based on the serving component (the audit source location) as either host-based, network-based or a combination of both. The host-based is usually located in a single host while the network-based system is usually located on machine separate from the hosts that it protects. Hybrid intrusion detection systems combine both the network and host-based systems (Alenezi and Reed, 2012).

There are two general forms of DoS attacks: those that crash services and those that flood services. DoS attacks are implemented by either forcing the targeted computer to reset, or consuming its resources so that it can no longer provide its intended service or obstructing the communication media between the intended users and the victim so that they can no longer communicate adequately (Silica Kole, 2013).

**1.2 STATEMENT OF THE PROBLEM**

Firstly, with the relatively immature network infrastructure, many network operators don't have the ability to inspect network traffic well enough to distinguish DDoS attacks from harmless traffic. Secondly, gateways that link IPv4 and IPv6 must store lots of ‘state’ information about the network traffic they handle, and that essentially makes them weaker and breakable. Divers challenges has been encountered in the network environment, where attackers spoof source IP addresses, and send out an indefinite quantity of packets attack that is above the average size or magnitude of IP addresses space. which consumes bandwidth, memory, CPU cycles, and any other resource that is necessary for normal operation. Due to the fact that IP’s occupies such a relatively small space, Internet security implementations are not taken into full consideration. This leaves a lot of networks vulnerable to various DDoS attacks. DDoS (such as SYN Flood) attack has posed a lot of threat on IP’s.

Various algorithms and models has been used to effectively address this problem. It is very important to develop a system capable of detecting various forms of attack on IP’s. Neural systems have effective learning calculations, and had been introduced as a contrasting option to computerize the improvement of tuning fuzzy frameworks. Neural systems present its computational attributes of learning in the fuzzy frameworks and get from them the translation and clarity of frameworks representation. This project work will make use of a model and algorithm to effectively address these situations.

**1.3 MOTIVATION**

The motivations for this research study are:

1. There is a need to adequately address and examine communication interrupt caused by various DDoS attack (such as SYN Flood) on Internet Protocols (Ipv4 and Ipv6) between client and server on a network, because it has posed a lot of threats and damages on the system as a whole.
2. Due to attacks on network of user Internet protocols, there is a need to gain adequate knowledge on network attacks and how to address those attack issues, using a more efficient security technique and methodology.
3. IP address spoofing that allows denial of service attack needs to be addressed to protect and help maintain the performance of computer systems and to protect information.
4. And then of the neural networks (ANFIS) have learning capacity, generalization capacity, and also very efficient.

**1.4 AIM AND OBJECTIVES**

The project aim is to simulate and detect DDoS (TCP SYN) Flooding attacks on IPV4 and IPV6 using an ANFIS model and Neuro-Fuzzy algorithm to compare the performance analysis.

**OBJECTIVES**

1. To use an ANFIS model and Neuro-Fuzzy algorithm to detect DDoS attacks on IPv4 and IPv6.
2. To gather, pre-process, train and test data for the experiment in (i).
3. To implement a protocol that will be helpful to detect and trace back the source of DDoS attacks on IPv4 and IPv6.

**1.5 CONTRIBUTION TO KNOWLEDGE**

This research work will make use of the proposed system to assist with prompt and accurate detection of DDoS attack on Ipv4 and Ipv6 so as to be able to ascertain the performance analysis of various network traffic and able to deduce the most suitable protocol for a particular network.

**1.6 PROJECT ARRANGEMENT**

**Chapter one:** Is the introduction of the project and it comprises background, statement of the problem, motivation, project aim and objectives, project methodology, contribution to knowledge and definitions of some terms used.

**Chapter two:** Contains an extensive literaturereview on various DDoS attacks. This will provide an in-depth knowledge of how to mitigate various form attacks.

**Chapter three:** Contains research methodology and it comprises requirement specification, analysis, design and also contains UML (Unified Modelling Language) diagrams that describes how the system works.

**Chapter four:** Contains the implantation procedure which consist of screen shots of the results and detailed discussion on how each component of the system works.

**Chapter five:** Conclusion of the work and proffers recommendation.

**CHAPTER TWO**

**LITERATURE REVIEW**

**INTRODUCTION**

A Distributed Denial of Service (DDoS) attack uses many computers to launch a coordinated DoS attack against one or more targets. Denial of service attacks and distributed denial of service attacks have become the network of one of the main critical problems. (“According to WWW FAQ regarding the security issues”).

Within seven years since its inception, the Internet had spanned across hundreds of networks across USA and Europe. By 1994 it reached over 3 million computers in 61 countries. This exponential growth continued and by 2001 there were 140,000,000 hosts (computers) linked to this massive heterogeneous network. With TCP/IP as its backbone, the Internet has a vast number of applications to offer like electronic mail, remote login and file transfer. The revolutionary application, which brought the Internet into public consciousness, is the World Wide Web (WWW). Like most other networking protocols, the TCP/IP is made up of different layers, with each layer responsible for a different role in data communication. TCP/IP Protocol Suite is normally considered to be a 4-layer system. This global network, the Internet, has become an integral part of worldwide economy and life of individuals (Virgeniya and Palanisamy, 2013).

DDoS attacks nowadays are part of every Internet user’s life. They are happening all the time, and all the Internet users, as a community, had some part in creating them, suffering from them or even loosing time and money because of them. DoS attacks do not have anything to do with breaking into computers, taking control over remote hosts on the Internet or stealing privileged information like credit card numbers (Mukhopadhayay et all, 2014).

One of the key elements in DoS detection technique is the time of detection (Peng et al, 2016). A good detection mechanism should detect the DoS attack before the service starts to be devalued. However, packets from an overloading type of DoS are often indistinguishable from those of legitimate users. According to the analysis made known by Mirkovic and Reiher (2004) that the detection of attacks is frequently difficult and increases the chance for a false positive, which is a critical problem in DoS detection. A good detection technique should react quickly and have a low false positive rate. Virgeniya and Palanisamy (2013) also outlined the attacks and performance factors of IPv4 andIPv6 protocols. In their work, they investigated the performance related metrics like throughput, delay, jitter and CPU usage are empirically measured on a test bed implementation. Protocols based on the performance evaluation were provided as an outcome the countless features. A small network of computing devices that started as Advanced Research Projects Agency (ARPANET) project is now a worldwide network of devices for most of users. Their work wedged the attention of many researchers and a board called Internet Control and Configuration Board (ICCB) was set up to co-ordinate the work. The ICCB came up with a set of standards that specified the details of how computers communicate and also a set of conventions for interconnecting networks and routing (transferring) data among them, technically referred as the TCP/IP Protocol suite, this could be used to communicate across any set of interconnected networks. The global Internet thus began during the early 1980’s when ARPA started using TCP/IP protocol on computers.

The work presented by Peng and Yonghua (2007) provided a general classification for DoS detection techniques. Both of the works have different naming for the classifications and are coinciding in the techniques stated. The first work was proposed by Peng, that general DoS classification covers two presented works. The detection methods in this work were divided into: DoS attack specific detection and anomaly-based detection. DoS attack specific detection covers a general and wide range of different detection techniques types under one classification. In anomaly-based detection, the techniques are based on a comparison between the network traffic and a prepared normal traffic profile. The second work was proposed by Yonghua. The detection techniques were divided into two types: IP attribute-based and traffic volume-based. The IP attribute-based technique monitors the behavior of selected IP attributes and considers the anomalies as deviation. The traffic volume-based studies the traffic of the network and applies statistical calculations on the packet rate of network flow.

Mukhopadhayay et all, (2014) also recognized different kinds of DoS attacks and then proposed a new methodology to simulate those attacks using MATLAB and Xilinx. And finally tested the network with some sample input data set to obtain the desired output data and accordingly, if fluctuation arises then we can decide if DoS attack has occurred in the network b ringing about successful DoS attack identification. This methodology was developed for comprehensive study of internet DDoS attacks, and made use of the agent-based approach. It was stated in their work that the DDoSSim enables one to deeply investigate different forms of attacks and protection schemes. The tool has the ability to provide useful recommendations on selecting best protection method. A neural network model follows a number of systematic procedures, which include: data collection, data processing, building the network, and training network.

Meena and Trivedi (2012) examined the tradeoff between different existing IP Trace back techniques and designed a novel protocol for IP trace back to detect DDoS attack based on Secure-Neighbor. They took the parameters of Response1 and Nonce from Secure Neighbour protocol and developed a novel protocol to find out the attacker's IP address at the moment the attack is taking place. The novel protocol applies the decryption function on Nonce and value of neighbour timeout of a particular node to find the metric value at the moment the attack is taking. They formulated the approach mathematically and solved each step of finding the IP address of an attacker for all possible entities. By using the Secure-Neighbor protocol every node has the information of all other nodes which are connected to that node and every node update the information of all its connecting nodes for every t-second. Furthermore, they developed the novel protocol for IP trace back to detect DDoS attack on a single interface model only.

Alenezi and Reed (2012) presented an introduction to intrusion detection systems (IDS) and survey of different DoS/DDoS detection techniques. The key observation in their survey is that a CUSUM-based detection technique has many advantages over other statistical instruments in that it is nonparametric; consequently, it does not require training and is more robust to variations in the attack profile. A signature-based detection is based on certain known characteristics in the traffic. Kompella (2004) mentioned that it is difficult to create a signature for a DoS attacks as the attackers could modify the type and the content. Furthermore, Cheng (2006) stated that signature-based detection could be used to detect the communication between the attackers and their zombies. However, the communication could be encrypted making this detection difficult. Subsequently, Peng (2007) stated that signature-based detection is incompetent for DoS detection. However, the signature-based should not be dismissed for the following reasons. First, although it is difficult to create signatures for all types of DoS, this fact applies to IDS more generally and not specifically DoS. There are certain types of DoS attacks that are straightforward to detect with a signature-based technique such as a TCP mixed flags attack. Secondly, Cheng noted that during the study one particular attack tool (Stacheldraht) that the communication between the attackers and the zombies can be detected using a signature based approach. This is highly useful at the prevention stage.

**2.1** **HISTORICAL BACKGROUND OF IPV4 AND IPV6**

According to: [http://www.arin.net.com](http://www.arin.net.com/) a Network in the world of computers is said to be a collection of interdepended hosts, via some shared media which can be wired or wireless. A computer network enables its hosts to share and swap the data and information over the media which can be connected across cities and colonies. According to “Wikipedia” Internet Protocol is a set of technical rules that conclude how computers relates over a network.

There are currently two versions: Internet Protocol version 4 (IPv4) and Internet Protocol version 6 (IPv6). The prevailing Internet Protocol standard is IPv4, which dates back to the 1970s. There are well-known limitations of IPv4, including the limited IP address space and lack of security (Durdagi and Buldub, 2010).

IPv4 is the first version of Internet Protocol to be widely used, and accounts for most of today’s Internet traffic. There are just over 4 billion IPv4 addresses. In Mobile IPv4, a node that receives the data packets resides on the specific network nominated to it by its corresponding IP address. IPv4 is the most demand addressing protocol used on the Internet and most individual networks today. With the advent of wide variety of devices and upcoming technologies, the limited addresses of IPv4 are not capable to handle with the current internet. IPv6 was mainly developed to resolve the addressing issues as well the security concerns which are lacked by IPv4 (Chauhan and Sharma, 2014).

IPv6 Internet Protocol version 6 (IPv6) is a new generation protocol of the basic internet protocol. Internet Protocol (IP) is a common language of the Internet, every device connected to the Internet must support it. IPv4 utilizes 32-bit address space while IPv6 utilizes 128-bit addresses. The major difference between IPv4 and IPv6 is the number of IP addresses. There are 4,294,967,296 IPv4 addresses. In IPv6, 340,282,366,920,938,463,463,374,607,431,768,211,456 IPv6 addresses, most networks that use IPv6 support both IPv4 and IPv6 addresses in their networks (Meena and Bundele, 2015).

**2.1.1** **IPV6 IMPROVEMENT OVER IPV4**

Ipv6 was designed to promote higher flexibility, improved security, better functionality and mobility support. IPv6 was designed with the principle of being able to coexist with IPv4 for a long period of time, and at the same avoiding breaking IPv4 networks, permitting all the services existing and applications to keep working without any interruption. At the same time, the way this coexistence work allows a smooth transition from IPv4 to IPv6. The basis of this coexistence and transition was to have both protocols in the hosts at the same time (dual-stack), and allowing the operating systems and/or applications to specify which protocol to be used for each communication (Virgeniya and Palanisamy, 2013).

The main advantage of IPv6 across IPv4 is address space. It was design to support +340 undecillion (2128) Internet Protocol addresses compared with 4.3 billion (232) IPv4 addresses. If the estimation of everybody in this world (6.77 billion) will require 3 IP addresses per person, then approximate the total required IP addresses for all the people around the world, which is 6.77 billion x 3 = 20.31 billion IP addresses (Khairil et al, 2009).

**2.1.2** **DENIAL OF SERVICE IN IPV6 NETWORK**

A DoS was made from a system or network while a DDoS attack is organized to happen simultaneously from a large number of systems or networks (Tixteco, et al, 2012). The denial of service attack, generated by utilizing the vulnerabilities in the network protocols, affect the performance of the victim as well as the other hosts sharing the network Meenakshi and Srivatsa (2007). Huang and Meng (2011) proposed an automatic model can be used to analyze the denial of service attacks in security protocol, so Meng protocol can be proved with a mechanized proof tool called ProVerif.

**2.1.3** **INTERNET PROTOCOL (IPV4 AND IPV6) ADDRESS SECURITY**

Network operators could provide higher service for every user with fine granularity source address filtered by protecting them from threats and tracing back the malicious host readily. Fuliang et al. (2011) observed that granularity source address validation has been deployed in various campus networks. In addition, the Duplicate Address Detection (DAD) algorithm was used to ensure that all configured addresses nodes on link are reliable. Guang et al. (2011) commented on a pull model called DAD which is used to improve a mechanism to secure DAD in IPv6, for the security needs of the Internet (IPv6), and based on the design by Yuhui et al. (2010) Net Flow data can be gathered to a traffic monitoring system to realize overall statistical analysis of the network traffic and thus alerts the system of the abnormal traffic. To find the MAC address in IPv6 networks, IPv6 uses Network Discovery Protocol (NDP). Barbhuiya et al (2011), demonstrated a technique for detecting spoofing neighboring solicitation and advertisement attacks.

Meenakshi and Srivatsa (2007) proposed a sequential method to detect DDoS attack fast which captures the increasing deviations from a normal performance at times. It has been found that although Internet Protocol Security (IPSec) provides security for IPv6 network, there is no security at absolute range. Therefore, the actual network Secure Sockets Layer or Time to Live (SSL/TTL) flow detection and other network technology should be combined together to protect attacks against potential threats (Hui, et al, 2011).

Virgeniya and Palanisamy (2013) empirically discussed the performance of two IP versions that has been. And the following conclusions were drawn from their discussion.

1. For packet sizes larger, IPv4 always gives a slightly better throughput than IPv6 However for small packet sizes the performance is almost identical.
2. The average jitter value is lower for the IPV6 protocol since it uses the connection oriented service and also employs the Plug and play connections on the network. It also selects the network dynamically.
3. The address space is larger for the IPv6 that improves the scalability for the users to locate the nodes on the network.

**2.2** **EXISTING METHODS FOR DDOS ATTACK DETECTION**

These processes are based on the architectures discussed above namely, victim-end, source-end and in-network. Past research works have shown that soft computing techniques have been employed largely for DDoS attack detection without applying the scheme towards VKC environment. It was discovered that a group of classifiers for DDoS attacks have also performed satisfactorily with high detection rates. DDoS attack detection can be classified into four major categories as follows:

1. **Statistical Methods:** Statistical properties can be applied for detection of DDoS attacks using normal and attack patterns. Typically, a statistical model with normal traffic method is adapted and then a statistical inference test is utilized to ascertain if a new instance conforms to this model. Processes that do not adapt to the learnt model, based on the applied test statistics, are classified as anomalies. Some authors have also proposed a method of Distributed Change Point (DCP) detection framework using change aggregation trees (CATs).
2. **Soft Computing Methods**: Learning paradigms, for example neural networks, fuzzy logic, neuro-fuzzy algorithm, radial basis functions and genetic algorithms are extensively applied in DDoS attack detection as a result of their potency to classify intelligently and automatically. Soft computing is a general term for identifying a collection of optimization and processing styles that are permissive of imprecision and uncertainty. Jalili (2005) introduced a DDoS attack detection system known as SPUNNID which is based on statistical pre-processor and unsupervised artificial neural nets. They used a statistical pre-processing to extract features from the traffic, and an unsupervised neural net to analyse and classify traffic patterns as either a DDoS attack or normal.
3. **Knowledge-based methods:** From this method, network events are analysed against predefined rules or patterns of attack. The approach gives general representations of known attacks which are formulated to identify actual occurrences of attacks. Examples of knowledge-based approaches include signature analysis, self-organizing maps, and state transition analysis, expert system.
4. **Other data mining and machine:** learning methods: An effective defense system to protect network servers, network routers, and client hosts from becoming handlers, zombies, and victims of DDoS flood attacks is presented in Hwang et al, (2003). The Net-Shield system protects any IP-based public network on the internet. It uses preventive and deterrent controls to remove system vulnerabilities on target machines.

Adaptation techniques are used to launch protocol anomaly detection and provide corrective intrusion responses. The NetShield system enforces dynamic security policies. NetShield is especially tailored for protecting network resources against DDoS flood attacks. Chen et al, (2007) presents a comprehensive framework for DDoS attack detection known as DDoS Container. It uses a network based detection method to overcome complex and evasive types of DDoS attacks. It works in inline mode to inspect and manipulate ongoing traffic in real time. By continuous monitoring of both DDoS attacks and legitimate applications, DDoS Container covers stateful inspection on data streams and correlates events among different sessions. It proactively terminates the session when it detects an attack.

**2.2.1** **ALGORITHMS AND TECHNIQUES FOR DETECTING DOS/DDOS ATTACKS ON NETWORK SERVERS AND INTERNET PROTOCOLS**

According to CSI survey in 2007, DDoS attacks were recognized as one of the major causes of financial losses. Raj and Selva (2013) used classifier’s combination for the detection of DDoS attacks. They also proposed an algorithm called NFBoost in terms of weight update distribution strategy, minimize the cost of errors and results combination method, the output of the has differ with other available methods. Vysrz et al (2014) responded to a new type of vulnerability emerged called the application layer denial of service attacks that target Web services, attack productive tools to test and confirm the reported vulnerability was introduced. Their results showed that, attacks devastating impact on the availability of Web services even when the absolute minimum of attack resources used. Because this type of attack is very simple to set up, it is clear that given the growth of cloud and web services, it is necessary to defend against them.

Aspydvrvpvlvs et al (2013) applied the method of game theory which is already in the field of network security, in order to explore the interaction between attacker and defender during the scenario of DDoS. The previous work with this model created a richer set of available choices to attackers, such as: the permutation of multiple writers in term of the cost to carry out an attack, the number of nodes being attacked, probability distributions of malicious traffic and they showed a unique optimal strategy available to the defendant by adopting this strategy.

Zhou et al (2014) suggested a new method to detect DDoS attacks of application layer (AL-DDoS). Their work differentiated itself with previous methods according to attack detection AL-DDoS. As the backbone of heavy traffic, they provided a modular architecture for defense that consists of manual end sensor, a detection module and traffic filter. They also proposed this method to build a Real-time Frequency Vector (RFV) and traffic real-time specification as a set of models developed. This model can be used to diagnose AL-DDoS attacks.

A DDoS detection technique was proposed, which is based on the source IP address. The system monitors the new source IP address of the packets instead of monitoring the traffic. The technique is based on the study by Jung et all (2002), which indicates that during an attack, most of the source IP addresses are new. On the other hand, during flash crowds most of the IP addresses are not new. A flash crowd is a dramatic increase in the load on a web server by a legitimate, large traffic surge causing an increase in congestion and packet loss. The main drawback of this technique is that the attacker could launch a DoS attack by known IP addresses to the target, in order to avoid the detection system. The attacker can start normal communication with the target and then perform the attack.

Talpade (1999) proposes a detection technique based on the characterization of the dynamic statistical properties of the network traffic such as TTL and other IP header information to detect the anomaly in the traffic. The characteristic of this idea is based on the change in the statistical distribution of the TTL values which indicates an anomalous change in the traffic. The main drawback is that the change in the TTL values does not always associate with anomalous traffic. Also, the model was not specifically proposed for DDoS.

Narayan et al, (2009) proposed various performance issues such as the delay in the network transfer, the management of the CPU time in efficient method for the diverse type of the operating system is provided. Furthermore, Che et al (2010) proposed the interoperability of the IPV6 protocol with the earlier versions, the transition mechanisms, the security aspects, the cost based on the speed of the protocols was discussed.

The research made by Limwiwatkul and Rungsawang (2004) discovered the DDoS attacking signature by analyzing TCP/IP packet headers against the well-defined rules and conditions, and distinguished the difference between normal and abnormal traffic. They were able acquire some sample signatures of the basic DDoS attack. **Kato and Klyuev (2014)** proposed an anomaly based DDoS detection system using radial basis function (RBF) neural networks. The proposed system was examined using a dataset from UCLA, and the results indicated that the system could attain real-time DDoS attack, and the detection accuracy was more than 96 %. Zhang et al (2009) discussed a prediction based detection algorithm against DDoS attacks, based on the proposed prediction, they were able to detect abnormal states of the server that may be initiated by DDoS attacks. Navaz et al (2013) proposed a statistical approach using joint entropy for DDoS attack detection and they also used the CAIDA dataset containing information for denial of service attacks. Their study calculated entropy for packets, with IP addresses, ports, and flow size as inputs, and compared normalized entropy with that of the assigned threshold value. The threshold value depends primarily on the false positive rate. Zhong and Yue (2010) analyzed the characteristics of DDoS attacks and the DDoS attack detection method using a data-mining algorithm. The fuzzy c-means (FCM) clustering algorithm and theory based association algorithm were used to extract features in network traffic. Bhuyan et al (2013) presented a survey with the results of DDoS attacks, detection methods, and tools used in wired networks.

Kim (1999) proposed a detection technique based on the creation of a stable baseline profile to monitor the abnormalities in the traffic. The analysis was conducted to check the stability of the traffic with regards to different parameters. Therefore, a baseline profile that is based on different attributes was proposed for detection. The choice of the attributes was based on the assumption that some of the attributes such packet size, TCP flag pattern, and protocol types can be anticipated by the attacker. Also on the other hand, Multi-Level Tree for Online Packet Statistics (MULTOPS) was proposed by Gil and Poletto (2001). It was a heuristic and data structure-based technique used by routers to detect a DDoS attack. The packet rate statistics for subnet prefixes are maintained by the nodes of the tree. According to them, MULTOPS encounters some limitations. The location and set up of MULTOPS routers in the network would affect the ability of the technique to detect attacks with randomized IP source addresses packets. Legitimate packets for a certain IP destination address will be dropped as the MULTOPS would be confused by the spoofed IP address packets and identify the destination address to be under attacks. Moreover, enormous number of attackers could connect to the victim in a normal way and the attacker’s flows rate of’ traffic is still proportional which means MULTOPS will not detect the attack. A detection technique for SYN flooding was proposed by Wang et al (2002). It is based on the normal behavior of TCP protocol (i.e. handshaking process and FIN or RST) and the sequential change point detection. The sequential change point detection is a statistical method to check for a change in a data. In order to make the technique easy to use and more general, a non-parametric cumulative sum (CUSUM) method was used. The technique compares the ratio of SYN packets to the FIN or RST to find a change. One of the drawbacks of this technique that is the attacker could send the FIN or RST along with the SYN packet to avoid the detection. The technique was based on statistical analysis on the data from different network layers to detect a change was proposed by Blazek (2001). The technique consists of two methods: adaptive sequential and batch sequential. The technique is based on the change point detection theory. To achieve a fixed rate of false alarms, statistical analysis of training data was utilized by both methods. The authors claim that their technique has three features: the methods are self-learning; the attacks can be detected with small delay; and computational complexity is manageable. The technique uses different traffic types such TCP and UDP in change detection modeling. The main drawback of the technique is the high computational complexity. Cheng (2006) proposes a technique, which is based on spectral analysis, to differentiate between normal traffic and attacker traffic. In order to use the spectral analysis in a packet-based network, a signal was defined as the number of arrival packets in a fixed length time interval.

The power spectral density of the signal was estimated to discover the periodicity. Based on the fact that the periodicity around the round-trip time of the normal TCP flows is strong in both directions while the attack flows are not, the attack is detected. The technique is not able to detect any attack other than TCP flows. Other protocols such UDP would pass undetected by the technique. A sophisticated attacker can send attack traffic at the same periodic interval to avoid detection such as Low-DoS. The attack traffic does not have to be from a single source to form high volume. An attacker could use the zombies to send normal behavior traffic to the victim. However, the large number of zombies would be enough to overwhelm and deny the service from the victim.

Furthermore, Kulkarni (2001) proposed a detection technique, which was based on Kolmogorov complexity, to detect DDoS attacks. Kolmogorov complexity calculates the size of the smallest representation of the data and measures the degree of the randomness. In general, it is based on the correlation between the traffic flows to distinguish between the attack traffic and high legitimate load traffic. It is being assumed that during the DDoS attack, the generated packets tend to have similar characteristics such as protocol type, destination address, type and execution pattern. All of the attack packets from different locations will have the same destination address which gives a similarity for the traffic pattern. This similarity can be detected by using the Kolmogorov complexity-based technique. On the other hand, the high load legitimate traffic tends to contain different types and characteristics which make the flow of traffic to be randomly distributed and not greatly interrelated. The technique was based on correlation and assumptions which are not always would be valid in case of the attack as the attackers can create a random flow to avoid the detection.

Cabrera, et al (2001) proposed a technique to proactively detect DDoS by using a time series analysis. The correlation between the traffic behavior at both of the victim and the attacker is the basis for this technique. A normal profile is built in order to compare any deviation in the traffic from the normal behavior to signal an attack alarm. In order to build the normal profile, key variables and correlation process need to be identified. Key variables are extracted at the  
victim side and then the variables, from the attackers, that are correlated to the extracted key variable are calculated by statistical tools such as Granger Causality Test (GCT) and Auto Regressive Model (AR Model).

In 2014 **Kato and Klyuev** developed an intelligent detection system for DDoS attacks by detecting patterns of DDoS attack using network packet analysis and utilizing machine learning techniques to study the patterns of DDoS attacks. In their study, they analyzed large numbers of network packets provided by the Center for Applied Internet Data Analysis and implemented the detection system using a support vector machine with the radial basis function (Gaussian) kernel. The developed detection system is accurate in detecting DDoS attacks. We calculated Bit Per Second (BPS) for all packet data and tried to detect the point when the attack started and difference in the amount of bytes between normal communication and abnormal communication. The amount of BPS for the attackers and victims.

**2.3** **REVIEW OF ADAPTIVE NEURO FUZZY INFERENCE SCHEME (ANFIS)**

Kotenko and Ulanov (2006) employed the use of a software simulation tool the DDoSSim which has been developed for comprehensive study of internet DDoS attacks, they repeated that the DDoSSim enables one to deeply investigate different forms of attacks and protection schemes; the tool has the ability to provide useful recommendations on selecting best protection methods. They make use of the agent-based approach; furthermore, they conducted experiments for protection against DDoS attacks in order to demonstrate some potentials of the DDoSSim. Moreover, they considered the different phases of defense operations, which include the learning, decision making and protection. They further investigated into the adaptation of these protection methods to the actions of the attacker(s). They also suggested a common approach and simulation environment for finding adequate defense methods against DDoS attacks, Attack and defense methods they used include: the attacker which could be a Daemon or a Master, on the other hand, the defense agents are categorized into: initial information processing (sensor), secondary information processing (sampler), attack detection (detector), filtering (filter), and finally the investigation (investigator).

Kotenko et al (2003) re-emphasized the fact that in order to fight DDoS attacks there is the need for fully understanding the theoretical basis upon which we can protect systems against such attacks, they propose an agent based framework for simulating and modelling DDoS attacks. Furthermore, they presented of a formal specification of a representative spectrum of DDoS attacks; finally, they implement an agent based software tool that has the potential of simulating DDoS attacks and responses of victim systems.

TCP/IP systems are unpredictable and stochastic in nature due to a number of factors such as constantly changing propagation channels, random mobility of users and sudden changes in network load. This renders conventional mathematical tools less effective for system modelling of communication systems. Thus, communication systems can be best modelled by adopting soft computing which exploits the tolerance for imprecision, partial truth and uncertainty to achieve robustness, low solution cost and tractability. One of such soft computing platforms is the Adaptive Neuro-Fuzzy Inference System (ANFIS). ANFIS is an architecture which can serve as a basis for constructing a set of fuzzy if-then rules with appropriate membership functions to give the specified input/output pairs model (Jang, 1993).

The exposing of ANFIS as a neuro-fuzzy classifier to detect intrusions in computer network was experimented by Toosi et al (2006). In their work, they evaluated the performance of ANFIS in the form of binary and multi classifier to categorize system’s activities into the normal and the suspicious activities. They used three steps to get the result. The preprocessing is the first mechanism to map the symbolic values of protocol into integer values. The result in the first methods is input of the neuro fuzzy to classify activities of the system, and the last step is performance comparison, based on detection rate and false alarm rate. To extend this mechanism, two machine learning, artificial neural network and fuzzy inference system were used as intrusion detection system.

(Chavan et al, 2004) proposed adaptive IDS using two phases which are training the algorithm used snort and the execution. The training was aimed to minimize the number of false alarm. To achieve the purpose, they built a signature patterns that would help encounter vulnerability. With the input retrieved from TCP dump, they match patterns from learning output and the signature database.

**CHAPTER 3**

**PROJECT METHODOLOGY**

**3.0 INTRODUCTION**

This section of the project discussed about the project methodology which entails mathematical method, different computer techniques, data collection, representation of project work using design tool like architecture as well the requirement specification of the proposed system, the system analysis and its design.

**3.1 METHODOLOGY**

Methodology is a sequence of steps that is needed to develop a system. The methodologyto evaluate the system is an effort made to choose the most suitable evaluation methodology.

The procedure of developing a FIS using the ANFIS framework (a soft computing technique), to help detect and analyse the performance of distributed denial of service (DDoS) on Internet Protocols (i.e. Ipv4 and Ipv6).

The proposed method presents a detection approach that is more accurate and efficient in the detection of DDoS. The proposed system will achieve a better performance using Neuro-Fuzzy algorithm, the detection of the attack will be more accurate and efficient, because ANFIS is the combination of Fuzzy Logic and Neural Network.

**3.2 REQUIREMENTS SPECIFICATION**

The requirements needed to simulate and detect DDoS attacks on Internet Protocols using ANFIS model in MATLAB to compare the performance analysis.

**3.2.1** **FUNCTIONAL REQUIREMENTS**

The following were the broad requirements considered at the outset:

A DDoS mitigation system, which:

1. Can protect multiple targets at a time.
2. Can work with minimal deployment delays.
3. Can work at “line speeds”, ensuring minimal processing time in both normal/attack situations.
4. Can reduce reaction time by speedy attack detection.
5. Is capable of detecting new attacks on the network traffic; and
6. Is finally implementable in dedicated hardware.

**3.2.2 NON-FUNCTIONAL REQUIREMENTS**

1. Availability: Failure of any critical component can cause service interruption (e.g. the application crashes due to attack, the database storage device fails,).
2. Security: The system is capable to achieve a secure solution and ensure a secure mechanism so as to detect DDoS (SYN-Flood) attacks.
3. Confidentiality: The confidentiality of information is compromised if an attack is launched. He/she may then get to know information not intended to be available for that person.
4. Performance: It would be efficient not to use up the processor and the system memory. When an application performs as expected under average or peak load.
5. Usability: The detection tools in the system is expected to be user friendly.
6. Reliability: The project would be reliable providing the expected result.

**3.2.3 SOFTWARE REQUIREMENTS**

The specification required for both Windows (32-bit or 64-bit).

1. MATLAB R2012a
2. GNU plot (used to draw traffic conditions).
3. Java Development Kit
4. Windows Server 2008
5. Wireshark (for packet filtering)
6. Traffic Monitor
7. Microsoft Excel 2016

**3.2.4 HARDWARE REQUIREMENTS**

For effective operation of the simulation process in the system, the following minimum hardware specification are recommended:

1. The system should have a hard disk of at least 20GB, Flash drive and CD-ROM drive.
2. Pentium IV and above
3. Disk Space: The minimum disk space should be 1GB (for MATLAB only) or (3 GB and above for a typical installation)
4. Processors (for both Windows and Linux): Any Intel® or AMD x86 processor supporting the SSE2 instruction set.
5. The system should to be used should be equipped with 14” VGA or SVGA monitor (coloured).

**3.3 SYSTEM ANALYSIS**

Using a given input/output data set, the toolbox function ANFIS constructs a fuzzy inference system (FIS) whose membership function parameters are tuned (adjusted) using either a back-propagation algorithm alone, or in combination with a least squares type of method. This allows fuzzy systems to learn from the data they are modeling. The signal reach is also optimized using ANFIS with transmission power and threshold as input. Thus, signal reach is output of the Neuro-Fuzzy system and inputs are transmission power and threshold.

**3.3.1** **OVERVIEW OF VARIOUS DDOS ATTACK**

As shown in figure 3.1 below, DDoS attack is classified as an active attack, and can either cause bandwidth depletion or resource depletion. Furthermore, bandwidth depletion can either be a flood attack or amplification attack, example of attacks in this category are UDP, ICMP, SMURF, FRAGGLE attack. Also, resource depletion can be a protocol exploit attack or malformed packet attack, TCP SYN Flood attack is an example in this category.

From figure 1, SYN Flood, victim server, firewall or other perimeter defense receives (often spoofed and most often from a botnet) SYN packets at very high packet rates that can overwhelm the victim by consuming its resources to process these incoming packets.  In most cases if a server is protected by a firewall, the firewall will become a victim of the SYN flood itself and begin to flush its state-table, knocking all good connections offline or even worse, or even reboot.  Some firewalls in order to remain up and running, will begin to indiscriminately drop all good and bad traffic to the destination server being flooded. SYN floods are often used to potentially consume all network bandwidth and negatively impact routers, firewalls, IPS/IDS, SLB, WAF as well as the victim servers.

![A description...](data:None;base64,)

**Figure 3.1:** DDoS Attack Overview.

**3.3.2** **ANFIS (FIS) Structure and Parameter Adjustment**

A network-type structure similar to that of a neural network, which maps inputs through input membership functions and associated parameters, and then through output membership functions and associated parameters to outputs, can be used to interpret the input/output map.

The parameters associated with the membership functions changes through the learning process. The computation of these parameters (or their adjustment) is facilitated by a gradient vector. This gradient vector provides a measure of how well the fuzzy inference system is modeling the input/output data for a given set of parameters. When the gradient vector is obtained, any of several optimization routines can be applied in order to adjust the parameters to reduce some error measure. This error measure is usually defined by the sum of the squared difference between actual and desired outputs. ANFIS uses either back propagation or a combination of least squares estimation and backpropagation for membership function parameter estimation. ANFIS model structure is shown in Figure 3.2.

![A description...](data:None;base64,)

**Figure 3.2:** ANFIS Model Structure.

**Source: (Pragathi and Shetty, 2012).**

**3.3.3** **A NORMAL AND ATTACK SCENARIO**

A real-life DDoS (SYN flood) attack will be examined. A Transmission Control Protocol (TCP) connection sequence works on the “three-way handshake” principle, where a SYN request is sent to initiate a TCP connection with the host and must be answered by a SYN-ACK response from that host, and then confirmed by an ACK response from the requester. A SYN flood DDoS attack exploits this principle in the TCP connection sequence to initiate an attack. The requester sends multiple SYN requests, but either does not respond to the host’s SYN-ACK response, or sends the SYN requests from a spoofed IP address. The host system continues to wait for acknowledgement for each of the requests, as a result tying up its available resources. There comes a time when no new connections can be made until the previous requests receive acknowledgement. This eventually results in a DDoS attack. As shown in figure 3.3 below.

![A description...](data:None;base64,)

**Figure 3.3:** A normal scenario and A (SYN Flood Attack) Scenario.

![A description...](data:None;base64,)

**FIGURE 3.4:** A TYPICAL SYN-Flood Attack.

A “three-way handshake”, which is a reference to how TCP/IP connections work, is the basis for this form of attack. As shown in figure 2 and 3 above. The SYN-ACK communication process works like this:

1. First, a “synchronize”, or SYN message, is sent to the host machine to start the conversation.
2. Next, the request is “acknowledged” by the server. It sends an ACK flag to the machine that started the “handshake” process and then waits for the connection to be closed.
3. The connection is completed when the requesting machine closes the connection.  
   A SYN flood attack will send repeated spoofed requests from a variety of sources at a target server.

The server will respond with an ACK packet to complete the TCP/IP connection, but instead of closing the connection, the connection is allowed to timeout. Eventually, and with a strong enough attack, the host resources will be exhausted and the server will go offline. Thereby making IP address spoofing possible for the attacker of the system to gain access to system resources, and launch various Flooding attack to the target server, which will deplete the resources and exploit Internet Protocols.

**3.3.4 DETECTION OF DDOS ATTACKS USING ANFIS Model.**

IP flow is composed of IP packets arriving one after another. As the basic data carrying unit of Internet, IP packet holds the upper layer’s information and can be easily gathered and controlled. Effective IP flow based detecting features will be analyzed. A packet set which is composed of packets belonging to the same time interval of Internet, and all these packets have the same specific characteristics, these same characteristics are called keys.

**MATHEMATICAL MODEL (Detection Model)**

A group of commonly used keys are: Protocol, Source IP (SrcIP), Source Port (SrcPort), Destination IP (DestIP), Destination Port (DestPort). Protocol is the protocol used by the upper layer, SrcIP and SrcPort are the source IP address and the source port number separately. DestIP and SrcIP are the destination IP address and the destination port number separately.

Let y(t) denote a site total traffic, which is the number of bytes arriving at a site (or server) at time t. Hereby, y(t) can be divided into normal traffic n(t) and attack traffic a(t), where attack traffic is generated by attackers. Then y(t) can be abstractly expressed by the following equation 1 and 2.

(1)

According to the similar feature of network traffic, we can use statistic traffic, which came from a site under no attacks before detection, instead of normal traffic during detection. Let *N*(*t*) denote the statistic traffic. So, Equation. (1) can be rewritten as

(2)

Evidently, when a site is not attacked, a(t)  0, this is to say, When the site is under attacks, a(t) will rapidly increase to high level. Therefore, if we can get the value of a(t) during detection, it should be very easy to discover attacks. Unfortunately, we have no way to get a(t) directly during detecting attacks. However, y(t) can be captured with sniffer software conveniently. According to Eq. (1), if we can get the value of *n*(*t*), then the aforementioned problem can be solved simply. But *n*(*t*) is also unknown in a period of detection yet. Hence, how to get *n*(*t*) becomes an essential problem.

To this purpose, a lemma was introduced as follows:

**Lemma.** xi (i  1, 2, , n) are n independent y = x1 + x2 +…+xn. For large n (e.g., n30), the distribution of y approaches a normal distribution. This lemma is just the central limit theorem in probability theory (Papuilis and Pillai, 2002).  
**Theorem.** In normal state, if the number of a site users is invariant, then the distribution of *y*(*t*) approaches the normal distribution.  
**Proof.** According to the condition of the theorem, we can assume that the site server has m users.  
Hereby, y(t) can be expressed by y(t) = y1 (t) + y2 (t) + …+ yn (t)

where i1, 2, , m, yi (t) is the traffic generated by the i*th* user. In normal state, the site users are independent of each other, so their traffic *y1*(*t*), *y*2(*t*)…, *yn*(*t*) are naturally independent. In addition, the number of a site users is generally far greater than 30. Therefore, the traffic *y*(*t*) satisfies the condition of lemma 1, the conclusion of the theorem is true.

**Building detection model**

When the site is under attack, includes attack traffic, this leads to . Hence, the mean of *A*(*t*) is far greater than zero. We can get a detecting method: if *A*(*t*) yields normal distribution with mean zero, we can determine if the server is secure, otherwise, there are attacks (Muhai Li, Ming Li and Xiuying Jiang, 2008).

A model for detecting attacks with the parameters estimate method of probability theory.

Let *T* and be the number of samples and the mean of random variable *A*(*t*) respectively, *u*(*T*) is the sample mean of *A*(*t*) with *T* samples. For the variance of *A*(*t*) is unknown, in order to estimate the mean η, we form the sample variance *S*(*T*):

In fact, the (*T*) is an unbiased estimate of the variance of *A*(*t*) (Papuilis and Pillai, 2002). Thus, under the assumption that *A*(*t*) is normal,

the ratio has a *Student*-*t* distribution with *T*-1 degrees of freedom.

Using the distribution, we can estimate the mean *η*. If we have known the confidence coefficient *P*, then *η* yields the approximate confidence interval where *δ*=1-*P*, and are the percentiles of the *t* distribution respectively. Appling actual data to this model, we discover, if the site is not under attack, the confidence interval of *η* is included in (-0.5, 0.5) or whatever range that is being gotten depending on the Confidence Interval of the Statistic traffic gotten). Otherwise, the relation above is not true. Thus, a model for detecting DDoS attacks is obtained.

Using the model above, the run-time detecting algorithm as follows:

1) Assign *P* and *T* an initial value respectively. The starting time of detection is 0.  
2) Open a database, which has stored statistic traffic of the site. Fetch data from the database

and load the data into array *N*(*t*); These data correspond with the time from 0 to *T*1.  
3) Set *u* (0)  *N* (0); *S*(0)  *y*(0), where *y*(0) is the traffic datum at starting time 0.  
4) Judge whether the relation *t* ≥ *T* is satisfied. If the answer is true, go to 8).  
5) Capture the traffic of the site at time *t*, and load it into *y*(*t*).  
6) Compute *u*(*t*) and *S*(*t*).  
7) Let *t* = *t*  1, and go to 4).  
8) Compute the confidence interval of *η*, this is .  
9) Judge whether the confidence interval of η is included in (0.5, 0.5), if the result is Yes,  
 then the site is safe; otherwise, gives an attack alarm.  
10) End.

**MEMBERSHIP FUNCTION USED FOR EACH OF THE INPUT VARIABLE**

In this section, the linguistic values are mapped to their respective membership functions using the triangular membership formula as:

**Triangular-shaped built-in membership function (trimf)**

**Syntax**

y = trimf(x,params)

y = trimf(x,[a b c])

**3.5.1** **EVALUATION METRIC**

Entropy is computed in order to detect the DDoS attack in a network. Entropy can be defined as measurement of the randomness and uniformity of the IP addresses. Entropy for the new prediction can be calculated as:

Where Whole Data (New Prediction), trained ANFIS data (Statistical traffic)

attack traffic, is the Confidence Interval, \* implies t distribution table.

**3.4** **METHOD OF DATA COLLECTION**

For conducting all the experiments, firstly the environment will be created using the data analysis tools for preprocessing of the real-time datasets. The preprocessing of data sets will be performed by using the traffic analysis tool (Wireshark 2.4.0.)

The simulation of this research will be carried out using MATLAB version R2012a. In this work, dataset will be collected, analyzed and pre-processed to the required format of ANFIS. The dataset obtained from the University ICT will be used as a training data while the remaining will be used as a testing data.

**3.5 DESIGN**

The simulation of a DDoS (SYN Flood) attack on Internet Protocols will require many ANFIS to be trained and tested in this research work to provide attack detection and classification in terms of their various performance. In addition, the result obtained will be used in ANFIS to categorize the datasets into attack and normal traffic using some standardized starting point values.

ANFIS model will be employed due to its ability to test and compute the performance analysis of DDoS attack. On the other hand, during flash crowds most of the IP addresses are not new. A flash crowd is a dramatic increase in the load on a web server by a, legitimate, large traffic surge causing an increase in congestion and packet loss, the system would be able to monitor the new source IP address of the packets and also to monitor the traffic flow and direction, in order to analyze the performance evaluation of the attack on Internet Protocols in which the data will be divided into both the training and testing dataset with the ratio of 70% to 30& respectively.

**3.5.2** **DESIGN OF PROPOSED ARCHITECTURE**

Designing neural network model follows a number of systematic procedures. In general, there are five basic steps shown in figure 3.5. The functionalities of this step are briefly described below.

![A description...](data:None;base64,)

**Figure 3.5:** Basic flow of designing artificial neural network model

1. **Data Collection:** First raw data is collected from trusted source to process for further steps. Data is collected according to the demand of the proposed network and dataset format should be compatible with the supported data format of the proposed network. Otherwise data set cannot be processed by the network.
2. **Processing Data:** This block takes the original data from the collected data source, extracts the required features, and converts the data set into MATLAB compatible format. This basically performs the data cleaning procedure.
3. **Building Network:** After the collection of desired data we have to create the proposed network model with the collected data set and some predefined useful parameter setting. The proposed network model parameters should be set by the user as per the required demand of the output feature.
4. **Training Network:** Training the network is a very important feature of DDoS attack simulation because here we make the network to behave the way as we proposed to i.e. we predefine the proposed network behaviour as our requirement i.e. to train it with the user defined way for its desirable behaviour.
5. **Testing Network:** After the training, the network is tested with some sample input data set to obtain the desired output data set and accordingly if fluctuation arises then we can determine that DDoS attack is happened in the network resulting in successful DDoS attack identification.

**ACTIVITY DIAGRAM FOR THE SIMULATION**

![A description...](data:None;base64,)

**FIGURE** **3.6:** The Activity Flow Diagram of proposed method.

As shown in figure 3.6 above the simulation procedures will be setup to evaluate DDoS attack, a SYN Flood attack in particular, the dataset obtained from the database (.csv file) will be pre-processed to suite the ANFIS supported format in form of a (.FIS) file. Data will be divided into training and testing dataset, and also used to give the performance evaluation using ANFIS toolbox collection in the MATLAB Integrated Development Environment (MATLAB IDE). This simulation will help to display the result graphically and mathematical based on the given parameters. Such that it will be possible to classify and determine the level of DDoS attack on the system.

**DETECTION FLOWCHART (ACTIVITY DIAGRAM FOR DDOS DETECTION)**

![A description...](data:None;base64,)

**Figure** **3.7:** DDoS Detection Flowchart.

![A description...](data:None;base64,)

**Figure 3.8:** Proposed Architecture for Network Traffic Analyzer

**CHAPTER FOUR**

**IMPLEMENTATION, RESULT AND DISCUSSION**

This chapter describes the implementation, design and result of the system developed, it entails the testing of the system, project schedule, project management, risk management, quality management and the various considerations taken when developing the system.

**4.1** **NETWORK SIMULATION**

The new defense mechanisms against DDoS attacks requires testing and validating the design exposition, which is not feasible by building the network and directly conducting a physical test because of the huge scale of the network involved. Also, this process requires careful attention and possibly research permission from authorities because DDoS attacks are illegal. Furthermore, the high cost associated with such test makes this testing technique almost impossible to follow, study, and develop the defense system.

A more attractive solution is to use a network simulation software package to simulate the performance of the network and the server system. Using this method, generating network designs produces safe, fast, reliable, and cost-effective results in order to study, enhance, and modify the design with relative ease. Wireshark Network Analyzer is the tool used to generate data obtained from the University ICT Unit.

The training (statistical) data used to train the network and results were obtained in an ANFIS output and the trained data served as input, until the minimum error was obtained to match the trained ANFIS model output, at 200 epochs the error was at its minimum state with a very high confidence interval (CI). In which the ANFIS Model now serves as a standard for monitoring network traffic, once the value (packet length) of the network traffic is greater than or equal to the “alarm value”, then DDoS is confirmed on network of either IPv4 or IPv6.

**4.2 TESTING**

The proposed model is tested with the predefined source and target dataset to obtain desired or fluctuation in result and also to determine the kind of predefined DDoS attack. The performance of the system taking into account training, validation and level 1 testing data. It shows that the best validation performance was 0.010563 at epoch 200. The input layer of the experimental setup has 112 input neurons to describe 112 attributes in the training data set.

After the data was cleaned, pre-processed and encoded it was fed to the neural network model of DDoS. The total time taken for completing the simulation was approximately 5 min. It took numbers of iterations to train the network.

The data used for training ANFIS (statistical) data was obtained from the University (FUOYE ICT) Unit, extracts the required features, and converts the data set into MATLAB compatible format. This basically performs the data cleaning procedure. The attributes given in the data set are converted into double data type to make it compatible with the ANN Tool box of MATLAB. Various features have been converted to variables of “protocol type” such as TCP, UDP, ICMPv6, ARP, STP, LOOP, MDNS, LLMNR and so on. “error flag” with corresponding values and parameters, such as: Numbers of data, Time, Source IP, Destination IP, Protocol Type, Packet Length and Information of each data respectively.

**4.2.1 TRAINING DATA**

The Neural network is trained with the given data set taken as source and target. Training parameters for ANFIS are taken as follows:

1. Size of the training data and testing data are 25084 and 17048 respectively
2. Number of TCP Data pairs extracted from the training dataset: 112
3. Number of epochs: 200
4. Number of nodes: 204
5. Number of linear parameters: 100
6. Number of nonlinear parameters: 150
7. Total number of parameters: 250
8. Number of fuzzy rules: 50

The proposed model is tested with the predefined source and target dataset to obtain desired or fluctuation in result and also to determine the kind of predefined DDoS attack.

![A description...](data:None;base64,)

**Figure 4.1 (a):** Comparison of Training Data and ANFIS data

The ANFIS training as at epoch 100 output with 20 membership function. The result above is not matching with the ANFIS output.

![A description...](data:None;base64,)

**Figure 4.1 (b):** Comparison of Training Data and ANFIS data

The statistical data has been trained to match ANFIS to attain the minimum error 0.580935 with step size of 0.010563.

![A description...](data:None;base64,)

**Figure 4.2:** ANFIS Training Data Error at each training epoch.

Some membership function was used for the input variable, the linguistic values are mapped to their respective membership functions using the triangular membership formula to set the input, fuzzy rules and output interpreters.

![A description...](data:None;base64,)

**Figure 4.3:** Root Mean Squared Checking Data Errors at each training epoch.

![A description...](data:None;base64,)

**Figure 4.4** Detection (Snapshot of the interface)

For improving the efficiency of detection, an alarm value was set. Once the traffic of the server reaches it, detection program will start automatically. In this work, the alarm value is 0.22×102; the length of detection time is 13.458 minutes, 14.947 and 22.21 with attack source; 172.217.17.141, 172.217.17.141, 172.217.17.36 respectively. Confidence coefficient *P* is 0.95; at different time interval. The detection program was executed and three of the sources gave attack alarm.

**Output Result and Evaluation**

Sample Mean of A(t): 1.5188

Standard Deviation S(T): 5.502

Percentiles of the t-distribution: -1.9816 1.9816

Attack Confidence Interval of DDOS attack is: 0.48857 2.549

The two Confidence intervals are not included in (0.5, 05), this means that the site was under attacks at 13.458, 14.947 and 22.21 minutes respectively.

**4.3 PROJECT SCHEDULE**

For the implementation of the system a network environment was supposed to be set-up, but due to the time frame and scope of the work, existing data was obtained from the school ICT Unit. Which include, the bandwidth management system for the school was used, and data were preprocessed, features extracted and data were tested and trained to for the implementation of the project work.

**4.4 QUALITY MANAGEMENT**

The basic goal of developing software is to produce quality software, the same applies to the development of this project which is to develop a reliable and dependable system. The functionality of this system as discussed in chapter three of this project were accomplished and tested, it was ensured that all the subsystems conforms to functional requirement of the system.

**CHAPTER FIVE**

**CONCLUSION AND RECOMMENDATION**

In this paper, by studying the basic feature of traffic, we give a model of detecting DDoS attacks. The model cannot be influenced by abrupt changes of normal traffic, and is independent of the starting time of detection. Implementation algorithm of the model with simple structure, low  
complexity, and low memory possession. With actual data to test the algorithm, the results show the algorithm can rapidly identify whether the server is under attacks. However, the detecting modal is dependent on statistic traffic before detection, the quality of the statistic traffic directly affects the result of detecting. Thus, it is very important to know the normal state of the site, and capture network traffic in time.

Conclusively, the purpose of this project is to get the performance evaluation of an attack on internet protocol, better understanding and more hands-on experience on Internet security was also ascertained. The simulation and detection of DDoS (SYN-Flood) attacks on IPV4 and IPV6 using an ANFIS model in the computation and performance analysis of a DDoS (TCP-SYN Flood) attack. Wireshark was used to gather data, while the data was pre-process, trained and tested using neuro-fuzzy IDE in MATLAB. An extensive test was also performed. Some preliminary results are reported using time and length of a TCP-SYN on the network. Also, ingress filtering is an efficient protocol to help trace back the source of DDoS attacks. In this research work, the Fuzzy Inference system model helps predict the corresponding dataset output values.

**5.1 CONTRIBUTION TO KNOWLEDGE**

1. Knowledge on how to apply ANFIS techniques to data modelling as seen from the other fuzzy inference GUIs.
2. Knowledge from simulating the detection and performance analysis of DDoS attack on internet protocols (Ipv4 and Ipv6).
3. Model Validation using training and testing sets. Model validation is the process by which the input vectors from input/output data sets on which the Fuzzy inference system was not trained, are presented to the trained Fuzzy inference system model, to see how well the Fuzzy inference system model predicts the corresponding data set output values.

**5.2 LIMITATIONS**

The limitation of this project was the technical difficulties while downloading the dataset, also the challenge encountered in gaining access to standard datasets, such as CAIDA 2007 DDoS attack, and NTUA 2003 DDoS attack. And it was difficult to get anyone to respond to various emails sent online about fixing the problem of getting those data. Consideration was made by creating a SYN Flood attack to get dataset manually, but ultimately decided that the University dataset will be large enough to perform a wide and robust test.

**5.3 RECOMMENDATION AND** **FUTURE WORKS**

Furthermore, only TCP-SYN attack was studied, future works should investigate other types of DDoS attacks on various networks.

**5.4 CRITICAL APPRAISAL**

This research work has helped in understanding the detection of various forms and types of DDoS attack that could attack a system, such that the user of the system will not have an idea of where to mitigate the attack because it tends to compromise lots of system by the use of botnets. Firstly, materials and resources that are not easily accessible and mostly not for free. Though most of the software’s used are open source and there was enough tutorials and related research work that helped the understanding of the background of the project work.

**BIBLIOGRAPHY AND REFERENCES**

A Navaz, V.Sangeetha, and C.Prabhadevi, “Entropy based anomaly detection system to prevent

DDoS attacks in cloud,” International Journal of Computer Applications, 2013.

A. Papuilis, S. U. Pillai, *Probability, Random* *Variables, stochastic Processes*, McGraw-Hill

Inc., 2002.

F. A. Barbhuiya, et al., "Detection of neighbor solicitation and advertisement spoofing in IPv6

neighbor discovery protocol," 2011, pp. 111-118.

G. Zhang, S. Jiang, G. Wei, and Q. Guan, “A prediction-based detection algorithm against

distributed denial-of-service attacks,” International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly, 2009.

G. Cheng, "Malware FAQ: Analysis on DDOS tool Stacheldraht v1. 666, "[Online]. Available:

http://www.sans.org/resources/malwarefaq/stacheldraht. php, 2006.

H. H. Bhuyan, H. J. Kashyap, D. K. Bhattacharyya, and J. K. Kalita, “Detecting distributed denial

of service attacks: methods, tools and future directions,” Oxford Journal, 2013.

IPv6: Current Deployment and Migration Status, by Xianhuiche, Dylan Lewis in International

Journal of Research and Reviews in Computer Science, June 2010.

J. Mirkovic and P. Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms," ACM

SIGCOMM Computer Communication Review, vol. 34, p. 93, 2004.

L. P. Tixteco, et al., "DoS Attacks Flood Techniques," International Journal, vol. 3, 2012.

L. Fuliang, et al., "Investigating the efficiency of fine granularity source address validation in IPv6

networks," in Network Operations and Management Symposium (APNOMS), 2011 13th Asia-Pacific, 2011, pp. 1-8.

L. Yuhui, et al., "Next Generation Internet Traffic Monitoring System Based on NetFlow," in

Intelligent System Design and Engineering Application (ISDEA), 2010 International Conference on, 2010, pp. 1006-1009.

L. Limwiwatkul and A. Rungsawang, “Distributed denial of service detection using TCP/IP header

and traffic measurement analysis,” International Symposium on Communications and Information Technologies, 2004.

Mukhopadhayay, S Polle, P Naskar. Simulation of Denial of Service (DoS) Attack using Matlab

and Xilinx. IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p- ISSN: 2278-8727 Volume 16, Issue 3, Ver. IV (May-Jun. 2014), PP 119-125 [www.iosrjournals.org](http://www.iosrjournals.org/)

Mohammed Alenezi and Martin J Reed. Methodologies for detecting DoS/DDoS attacks against

network servers. The Seventh International Conference on Systems and Networks Communications, ICSNC 2012. ISBN: 978-1-61208-231-8.

Performance Evaluation of IPv4 and IPv6 on Windows Vista and Linux Ubuntu by Shaneel

Narayan (Member IEEE), Peng Shang, Na Fan”. In 2009 International Conference on Networks Security, Wireless Communications and Trusted Computing.

R. Zhong and G. Yue, “DDoS detection system based on data mining,” Proceedings of the

Second International Symposium on Networking and Network Security, 2010.

R. R. Kompella, S. Singh, and G. Varghese, "On scalable attack detection in the network," in

Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement. ACM Press, New York, 2004, pp. 187-200.

S. Meenakshi and S. K. Srivatsa, "A Distributed Framework with less False Positive Ratio

against Distributed Denial of Service Attack." Information Technology Journal, p. 6, 2007.

S. Meenakshi and S. K. Srivatsa, "A Distributed Framework with less False Positive Ratio against

Distributed Denial of Service Attack." Information Technology Journal, p. 6, 2007.

S.Clement Virgeniya, Dr.V.Palanisamy. Attacks on Ipv4 and Ipv6 Protocols and it’s Performance

Parameters. International Journal of Computer Trends and Technology (IJCTT) – volume 4 Issue 8– August 2013. P. 2429.

T. Peng, C. Leckie, and K. Ramamohanarao, "Survey of network-based defense mechanisms

countering the DoS and DDoS problems," ACM Computing Surveys (CSUR), vol. 39, p. 42, April 2007.

W. Hui, et al., "DDoS/DoS Attacks and Safety Analysis of IPv6 Campus Network: Security

Research under IPv6 Campus Network," in Internet Technology and Applications (iTAP), 2011 International Conference on, 2011, pp. 1-4.

W. Huang and B. Meng, "Automated proof of resistance of denial of service attacks in remote

internet voting protocol with extended applied Pi calculus," Information Technology Journal, vol. 10, pp. 1468-1483, 2011.

Yogesh Kumar Meena1, Aditya Trivedi. A Novel Protocol for IP Traceback to Detect DDoS

Attack. IJCSI International Journal of Computer Science Issues, Vol. 9, p. 290, Issue 4, No 1, July 2012 ISSN (Online): 1694-0814. [www.IJCSI.org](http://www.IJCSI.org/)

“The 12th annual computer crime and security survey,”

<http://i.cmpnet.com/v2.gocsi.com/pdf/CSISurvey2007.pdf> [accessed: April 19, 2017].

[Online] Available: [http://www.arin.net.com](http://www.arin.net.com/)

[Online] Available: <http://www.google/wikipedia.com>.

**APPENDIX – USER MANUAL**

All implementation codes related to this project are inserted into the CD attached to the project report.