DETECTION OF FLOODING DISTRIBUTED DENIAL OF SERVICE ATTACKS IN RULE-BASED NETWORK INTRUSION DETECTION SYSTEMS

By

Amtul Saboor

A thesis submitted to the faculty of Department of Information Security Military College of Signals, National University of Sciences and Technology, Rawalpindi in partial fulfillment of the requirements for the degree of MS in Information Security

October 2014

ABSTRACT

DETECTION OF FLOODING DISTRIBUTED DENIAL OF SERVICE ATTACKS IN RULE-BASED NETWORK INTRUSION DETECTION SYSTEMS

by

Amtul Saboor

Distributed Denial of Service (DDoS) attack is launched by sending huge network traffic to a victim system, using multiple systems resulting in unavailability of services to legitimate users. Detecting such attacks has gained much attention in current literature. Studies have shown that flow-based anomaly detection mechanisms give promising results as compared to typical signature based attack detection mechanisms, which have not been able to detect such attacks effectively.

The thesis starts with an investigation of the detection techniques used by Rule-Based Network Intrusion Detection Systems for detecting flooding DDoS attacks. A variety of flow-based DDoS detection algorithms have been put forward for detection of flooding DDoS. The flow-based DDoS attack detection techniques have been divided broadly into two categories: Packet Based and Mathematical Formulation Based. Analyses has been done on two recent techniques one belonging to first category; IP Address Feature Value (IAFV) and the other belonging to second; Correlation of IP addresses.

In order to analyze the algorithms under study effectively, two different test benches have been established, one using real systems and the other using DETERlab. Both of the algorithms have been analyzed under several normal and flooding DDoS attack scenarios and evaluation has been done with respect to their detection capability and accuracy. The correlation technique has been found to outperform the rest of the techniques and has been finally chosen for further improvements by introducing multiple sliding time window intervals and calculating correlation coefficient for each of them. A comparison of correlation coefficient values over multiple sliding window time intervals leads to better decision making. The proposed technique was then implemented and integrated with the de-facto rule-based network intrusion detection system, Snort. The effects of the algorithms integrated with Snort were evaluated and results were generated to see the impact of the proposed technique. Finally, an analyses of the proposed technique has been conducted with respect to false alarms. It has been found that the proposed multiple sliding window correlation technique outperforms the old correlation technique and Snort's default flooding DDoS attack detection mechanism.

SUPERVISOR CERTIFICATE

It is to ceritfy that final copy of thesis has been evaluated by me, found as per specific format and error free.

Dated:_____ _____

(Lt. Col. Baber Aslam, PhD)

DECLARATION

I hereby declare that no portion of work presented in this thesis has been submitted in support of another award or qualification either at this institution or elsewhere.

_____

Amtul Saboor

DEDICATION

I dedicate this thesis to my parents, who supported me each step of the way.

ACKNOWLEDGEMENTS

In the name of Allah, the Most Gracious and the Most Merciful

All praises to Allah for the strengths and His blessing in completing this thesis. I would like to convey my gratitude to my supervisor, Dr. Baber Aslam, for his supervision and constant support. His invaluable help of constructive comments and suggestions throughout the experimental and thesis works are major contributions for the success of this research. Also, I would thank my committee members; Dr. Mehreen Afzal, Lecturer Mian Mohammad Waseem Iqbal and Lecturer Waleed bin Shahid for their support and knowledge regarding this topic.

I would also thank system administration team of MIS Cell of Military College of Signals, for providing me with the required MCS network related data. Also, I would like to thank Abdullah Noor Muhammad for his insightful conversations and helpful suggestions.

Last, but not the least, I am highly thankful to my parents (Riaz Ahmad and Amtul Hafeez) and siblings (Dr. Sobia Naz, Dr. Sadaf Hafeez and Dr. Shagufta Kanwal). They have always stood by my dreams and aspirations and have been a great source of inspiration for me. I would like to thank them for all their care, love and support through my times of stress and excitement.

LIST OF FIGURES

LIST OF TABLES

.

TABLE OF CONTENTS

LIST OF ACRONYMS

DoS.....Denial of Service

DDoS.....Distributed Denial of Service

NIDS.....Network Intrusion Detection System

LOIC.....Low Orbit Ion Cannon

RBF.....Radial Basis Function

LVQ.....Linear Vector Quantization

RBP.....Resilient Back Propagation

DPM.....Deterministic Packet Marking

PBM.....Probabilistic Packet Marking

TCP.....Transmission Control Protocol

UDP.....User Datagram Protocol

ICMP.....Internet Control Message Protocol

PCA.....Principal Component Analysis

HMM.....Hidden Markov Models

CPR.....Congestion Participation Rate

LDDoS.....Low rate Distributed Denial of Service

EWMA .....Exponentially-Weighted Moving

IAFV.....IP Address Feature Value

Chapter 1

Introduction

1.1 Introduction

Computer networks play substantial role in today's information technology oriented world. Recent statistics show that number of computer users in Pakistan is rising with time. Right now over 30 million people in Pakistan use internet[1]. According to the report, internet penetration in the country has reached 16%. Nearly all organizations use communication methods like computers, laptops, handheld devices and routers etc. This is to ensure that users can remotely gain fast and easy access to programs and databases within or outside the enterprise's network. An organization's performance is directly proportional to the fact that the necessary information is available at times when needed. Thus it can be concluded that the availability of required information is an essential aspect for progress of any environment. In this context, the major and the most destructive network attack in today's world is distributed denial-of-service (DDoS). This is an attack where multiple compromised hosts are used to target a single victim causing denial of service on that system. Such attack overwhelms all network servers and devices and service becomes unavailable. Detection of DDoS attacks is the primary focused subject of this thesis.

This chapter gives an overview of the basic concepts underlying behind this research such as DDoS attacks, flooding DDoS attacks, network intrusion detection systems (NIDS) and their types, explanation and limitations of rule-based NIDS and a brief overview of flaws in existing flooding DDoS attack detection schemes. After delivering the prerequisite knowledge and concepts, the aim, motivation, scope and contributions have been explained. Finally organization of rest of the thesis is given.

1.2. DDoS Attacks

The impact of a DDoS attack is directly on the availability of the information that is requested by the user. The information can be of any type e.g. web pages, online services like gaming or network bandwidth. Significance of availability of information is felt when it becomes unavailable even for a short time. This leaves a bad impression about the enterprises' reputation and many times results in business loss. The very first large scale and deliberate DoS attack occurred at the University of Minnesota in August 1999 that used bots collectively to flood victim machines. The attacks with similar effect still exist and are increasing day by day exponentially[2]. From last 2 years DDos Attacks have been among top 10 network attack techniques [3]. In 2013, a massive 300 Gbps DDoS attack was launched against Spamhaus' website. Later, in 2014, a 400 Gbps DDoS attack was launched against US and EU based servers [4].

1.2.1 Classification of DDoS Attacks

Since DDoS attack is a very generic terminology, various studies have classified the DDoS attacks into 3 broad categories[5][7][8].

1.2.1.1 Bandwidth-Depletion DDoS attacks: The main target of such attacks is consumptions of network bandwidth or resources and their depletion before a legitimate users might be able to use them, thus causing unavailability of the required bandwidth. Such attacks involve sending huge amount of data to network devices like routers, servers or firewalls to overload the network and cause denial of their services.

1.2.1.2 Volume Based/Flooding DDoS Attacks: This class of DDoS attacks is the most common and difficult to detect. It involves sending huge legitimate requests to victim servers, either by original or forged source addresses. In this way, the legitimate users cannot access the required services since the servers are busy in responding to the packets sent by attackers. Common attack traffic in such cases includes TCP, UDP and ICMP packets.

1.2.1.3 Application based DDoS Attacks: The main target of such attacks is the application layer which is the 7th layer in OSI model [9]. This is done by exploiting a known or zero day vulnerability in an application (most commonly in operating systems). A full TCP connection is established by the attacker just like legitimate user. Since an attack is launched after establishing a legitimate connection, such an attack is relatively difficult to detect but they are easy to defend against once detected [81].

1.2.2 Criticality of Flooding DDoS attacks

The criticality of detecting flooding DDoS attacks relies on the fact that anyone can use simple and easy to use free DDoS traffic generating tools available. Also, the packets that are part of the attacks do not contain any specific payload that can be matched

with pre-existing signatures. Few of the main target industries are media and entertainment, software and technology, security, financial services and gaming [10]. Besides, cyber statistics have shown that there has been 47% increase in total DDoS attacks over last year and 39% increase in average bandwidth of the attack in [11][12]. Among the various types of DDoS attacks, flooding DDoS attacks occur with highest percentage. The cyber statistics [97][98] show that there has been 718% increase in DDoS over 2013. According to the annual report about DDoS attack vectors and their distribution in 2014, the highest occurring attacks are of the flooding type that mainly included ICMP (9.82%) , Syn (17.69%) and UDP (10.36%) floods [10].

1.3 Network Intrusion Detection System

Network Intrusion detection system is a hardware or software that monitors network activities for malicious activities or policy violations and produces reports to a management station.

1.3.1. Types of NIDS

Various types of intrusion detection systems are there. They can be categorized broadly as rule-based (signature based), anomaly based (behavior based) and hybrid.

1.3.1.1 Rule-based detection:

Such intrusion detection systems compare incoming attack traffic to previously stored attack signatures derived on the basis of set of rules or attack patterns to identify occurrence of attack traffic. Rule-based intrusion detecting systems are able to detect known and commonly occurring DDoS attacks whose signatures are already present in

their database. An alarm is raised whenever a match is discovered. NSM, Bro, and Snort are the examples of rule based intrusion detection systems [6][18][22] . As compared to anomaly based systems, such a NIDS gives lesser false alarms but is unable to identify unseen and novel attacks like flooding DDoS attacks. This will be discussed in detail in Chapter 2.

1.3.1.2 Anomaly-based detection:

This type of network intrusion detection system collects data related to normal or legitimate users for a certain time period. Later on, alarm is generated whenever the incoming traffic is not matching with the normal behavior data set already collected. An example of such a detection system is MULTOPS [44], it uses heuristics to measure behavior deviation by looking at different incoming packet rates. PAYL and MCPAD are other examples of anomaly based NIDS [95] [96]. Such NIDS is a step ahead of signature based network intrusion detection systems in the sense that it has the ability to detect new attacks whose signatures or rules have not been known before. However, they have a higher chance of occurrence of false alarms. There are many ways to fine tune the results for reducing false alarm rates.

1.3.1.3 Hybrid Detection:

An intrusion detection system that involves using qualities both of an anomaly based and signature based system is known as hybrid NIDS. After examining different positive features of different anomaly based and signature based systems, this approach combines benefits of different NIDS belonging to both types. Studies indicate that this is

found to be a better approach as compared to both of them separately since it covers limitations of both.

1.4 Motivation and Problem Statement

Probability of occurrence of flooding DDoS attack incidents is rising with time and causing damage to individuals, websites, servers and network daily[10][97][98]. It has been used by hackers, hacktivists and cyber-terrorists because of limited detection mechanisms against it. Highly sophisticated and deceptive flooding DDoS attacks can bypass firewalls easily.

Rule-based detection is the most commonly used methodology to detect flooding DDoS attacks. The de-facto intrusion detection system, Snort is also based on the rule-based detection[75]. Unfortunately, it suffers from limitations as it cannot monitor traffic flow [94] and thus cannot detect flooding DDoS attacks efficiently as discussed in [13][14][15][16][17] .Literature has shown that most commonly used NIDS are short of detecting flooding DDoS attacks if used exclusively because they lack intelligent traffic analysis.

Since, rule-based NIDS, Snort is open-source and most commonly used, finding an appropriate countermeasure for flooding DDoS attacks and integrating it with Snort poses a great challenge for organizations worldwide. It is utmost need of today's growing dependence on internet to detect such attacks timely, accurately and efficiently. The problem statement is "**There is a need to explore and analyze the detection capability of flooding DDoS attacks in rule-based NIDS with the analyses of**

the extent to which existing techniques detect those attacks and introduce a flooding DDoS attack detection technique that outperforms the existing detection methods".

1.5 Flow Based DDoS Detection Techniques

Flow based detection technique is new enhancement to DDoS protection. A flow is a unidirectional data stream where all packets share some or all of these characteristics: IP source and destination address, source and destination port number and protocol value [20][62]. The idea of this technique is basically to use only a part of information from headers of incoming packets and analyze this header information by grouping the incoming packets in the form of flows. Studies indicate that flow detection is much more scalable than a solution relying on rule based signature database. Such a mechanism tracks all packets thus consuming memory resources much more that flow based mechanism [21]. Flow detection techniques consume lesser resources as they track only header information from the incoming packets . Also they have the ability to detect novel DDoS attacks better than payload based detection mechanisms [13][14][15][16][17][60]. Integrating such a method with Rule Based NIDS before its detection engine will make it much more proficient.

1.6 Aims & Objectives

This thesis aims primarily at achieving the following goals:

1. Analysis of existing flooding DDoS attack detections techniques that detect such accurately and reliably.
2. Development of efficient flooding DDoS attack detection method.

3. Integration of flow based DDoS detection techniques with rule-based NIDS, the flooding DDoS attacks that are missed by other means are targeted to be identified by adding the proposed capability.
4. Analysis of proposed method with respect to traditional detection technique used by rule-based NIDS generally is to be presented.

1.7 Thesis Contributions

This section explains the 4 major contributions from this thesis.

1.7.1 Evaluation of Snort Against DDoS Attacks under Different Hardware Configurations

During the experimental test bench set up phase, effort has been made to gauge Rule-based NIDS in terms of performance (packet handling) and detection accuracy against Flooding Distributed Denial of Service attack. The evaluation has been done using a sophisticated test-bench under different hardware configurations. Experimental results have shown significant improvement in packet handling capability by using better hardware. However; detection capability of Rule-based NIDS is not improved by improving hardware and is dependent upon its internal architecture (signature database and rate filtration). This research outcome led to the following publication:

"A. Saboor, M. Akhlaq, B.Aslam, "Experimental evaluation of Snort against DDoS attacks under different hardware configurations", In Proceedings of 2nd National Conference on Information Assurance (2013)"

1.7.2 Categorization of Various Flooding Based DDoS Attack Detection Techniques

While investigating various DDoS attack detection techniques, recent flooding based schemes have been studied in detail. The 2nd contribution of this thesis is the categorization of these schemes into two broad categories namely, packet header based and mathematical formulation based techniques. Chapter 2 explains each category along with the schemes belonging to each category as well as their limitations.

1.7.3 Traffic Generation

Network traffic generating tools have been used to generate and deploy flooding DDoS attack traffic and normal traffic that closely resemble real-world scenarios. A realistic traffic generation framework has been co-operatively developed as a part of this research in order to synthetically generate and deploy different attack and normal traffic scenarios. The framework makes use of modest hardware and exploits the random IP generation feature of various attack generating tools, which is used for sending network packets with multiple distinct IP addresses from a single source machine to a single destination machine.

1.7.4 Test-Bed I & II Formulation

In order to analyze the algorithms under study effectively, two different test benches have been established, one using real systems and the other using DETERlab. The main goal behind testing the algorithms using two distinct test-beds was to analyze the effect of change in test-beds in the performance of the algorithms and packet handling capacity of Snort.

1.7.5 Implementation of Improved Correlation Technique to Detect DDoS Attacks

The 3rd contribution of the thesis is the design and a proof-of-concept implementation of a DDoS attack detection technique based on correlation of the incoming packets over multiple sliding window time intervals. The proposed technique is an extension of the previous research conducted in this direction and has helped to minimize false negatives and false positives that are faced in the old scheme [55].

1.7.6 Integration of Improved Correlation Technique With De-Facto Network Intrusion Detection System

The proposed technique is integrated with the de-facto rule-based NIDS as a dynamic preprocessor. To the best of our knowledge, to date, no similar technique has been introduced within the chosen rule-based NIDS for detection of flooding DDoS attack.

1.7.7 Analysis of Improved Correlation Technique

Analysis of the proposed technique has been conducted with respect to false positive and false negative alarms. It has been seen that the proposed correlation outperforms the old correlation technique and Snort shows much better results in terms of detecting flooding DDoS attacks when the improved correlation algorithm is integrated with it.

1.8 Thesis Organization

The rest of the thesis is organized as follows:

In Chapter 2, mechanism for detecting flooding DDoS attacks incorporated in rule based NIDS is evaluated and their limitation has been discussed. Then the solutions to detect

distributed denial of service attacks in general and flooding DDoS attacks in particular, as proposed by other authors, will be presented and discussed.

In Chapter 3, flow based techniques explained in Chapter 2 have been classified into two categories namely, packet header based and mathematical formulation based. Then the two chosen flow based techniques, IAFV and Correlation techniques are explained in detail along-with mathematical explanations of both algorithms.

In Chapter 4, design of flooding DDoS attack detection technique based on improved correlation technique is presented. The proposed technique extends the previous work done in this direction by [55]. They analyzed correlation coefficient of incoming network packets per two consecutive intervals and observed that the value of correlation coefficient is abnormally reduced during attack conditions; since there will be larger set of unique source IP addresses per unit time. We propose multiple sliding window time interval correlation analyses using correlations of 4 consecutive sliding windows, in order to reliably determine if the current incoming network traffic represents attack condition or not.

In Chapter 5, the test scenarios, test-beds and implementation details have been explained. Snort has been chosen as the subject NIDS as it has achieved the position of de-facto standard among all the NIDS. Two test benches have been used, one comprising of physical systems called Test-bed 1 while the other is emulation based on DeterLab called Test-bed 2. While keeping the packet per second range steady, variations in the uniqueness of source IP addresses has been tested against both

algorithms and for both test-beds; 1 and 2. Tests have been done on attack traffic as well as on normal traffic.

In Chapter 6, results have been presented. It has been shown from results that the proposed correlation algorithm successfully identified the attack instances in all the attacks scenarios with very low rate of false negatives and no legitimate traffic was detected as attack traffic, hence, there were no false positives at all. Therefore, the proposed correlation technique outperforms the rest of the techniques.

In Chapter 7, concluding remarks have been given. The achieved objectives have been explained in detail. Besides, the limitation and future directions have been discussed.

1.9 Conclusion

In this chapter the basics of DoS and DDoS have been covered along with their types. Criticality of detecting flooding based DDoS attack has been thrown light upon. Then, types of NIDS have been discussed and the current status of rule-based NIDS in terms of detecting flooding based DDoS attacks has been explained. The main objectives, motivation, problem statement and scope of the thesis have been explained. In the end the organization of the rest of the thesis was described.

Chapter 2

Literature Review

2.1 Introduction

In this chapter, mechanism for detecting flooding DDoS attacks incorporated in rule based NIDS is evaluated and their limitation has been discussed in Section 2.2. Then the solutions to detect distributed denial of service attacks in general and flooding DDoS attacks in particular, as proposed by other authors, will be presented and discussed. The proposed solutions have been divided into three broad categories as depicted in Figure 2.1. In neural network based DDoS detecting schemes, the system under attack is trained using various techniques mainly Linear Vector Quantization, Radial Basis Function, Back Propagation and Resilient Back Propagation. Such a system has needs to be trained well for attack scenarios which in turn requires huge memory and CPU consumption as discussed. Trace back schemes mainly emphasize on finding that DDoS attack is occurring and then tracing the attack backwards with the aim of pinpointing the attacking source. Several recent schemes have been discussed along with their limitations. Statistical DDoS detection techniques aim at detecting attack packets by various statistical measurements or metrics of the composition of the traffic. Flow based detection is a type of statistical detection technique. Various latest flow based detection schemes have been explained along with shortcomings of these respectively. At the end, a summary of flow based schemes has been given in the form of table.

2.2 Flooding DDoS Detection Solutions In Rule-Based NIDS

In this section, solutions provided by rule based NIDS as a detection mechanism for flooding DDoS attacks are presented along with their shortcomings and limitations. Typical rule-based NIDS introduce a mechanism generically called rate filtration. At first, this seems to be a good way of detecting and limiting packets and thus, controlling DDoS attacks but a closer observation results in failure of this feature when the number of packets sent by attacker IP addresses are within the limits of its parameters. Typical rate_filter parameters of a famous rule based network intrusion detection system Snort is as follows [94]:

```
rate_filter \ gen_id 1, sig_id 469, \
        track by_dst, \
        count 25, seconds 60, \
        new_action drop, timeout 30
```

The above parameter asks to raise an alert if a network packet is received towards same destination IP with a rate of 25 packet per second and keep alerting for 30 seconds if the condition persists. If "count" parameter is further decreased to 10 or 5 per 60 seconds, there is a strong possibility of facing a high false negative rate as it treats legitimate packets as attack packets and drops them. Hence, it is very easy to bypass this feature which is the only feature in Snort to defend against flooding DDoS attacks. This has been shown in Chapter 6.

Cearns introduced a flooding DDoS detection preprocessor in an open source rule based NIDS that basically limited incoming packet rates. Later in 2002 Snort developer team included a rate filter parameter within its rules but it drops even legitimate packets after the rate exceeds a defined threshold [91].

Uyar et al. suggested that In order to detect DDoS attacks of various types, under various circumstances, signature-based and anomaly-based IDS should be hybridized [92] so that every type of attack is detected efficiently because all attacks cannot be detected using either of the algorithms exclusively.

Evaluation of rule based NIDS has been done against Low Orbit Ion Cannon (LOIC) attack in 2010. LOIC [82] is a network stressing and denial of service attack launching tool. It was found that the NIDS has not been effectively mitigating even this well known attack, although signature against LOIC is present. This is because the attack signature needs to be learned and provided to the NIDS in advance while the attack has thousands of participants. Every attacker may have specified its own content string, making this method not very effective. Rate filtering based on threshold does not distinguish between legitimate and attack traffic and thus, drops all packets [93].

In 2011, a rule based NIDS has been evaluated using realistic attack traffic with four DDoS attacks and results showed that at lower rate attack , NIDS performed

effectively but it completely failed when incoming packet rate was higher than 6000 packets per second [74].

2.3 Recent Flooding DDoS Detection Solutions

This section explains the major recent DDoS detection solutions that claim to detect flooding DDoS attacks. Broad categories of solutions are illustrated in Figure 2.1.

2.3.1 Overview of Neural Networks Based Solutions

Neural networks are designed to operate like a human brain. Neural network systems are set of programs that handle large number of procedures in parallel. Every single procedure has its own memory unit and knowledge base. System based on neural network is trained with huge data based on several principles and rules; that makes it able to behave spontaneously in case of real time novel situations.

2.3.1.1 Solutions Based on Neural Networks

Gavrilis and Dermatas in [23] proposed a DDoS detecting scheme in public networks using estimated statistical features and Radial Basis Function (RBF) neural networks for precise classification of the attacks

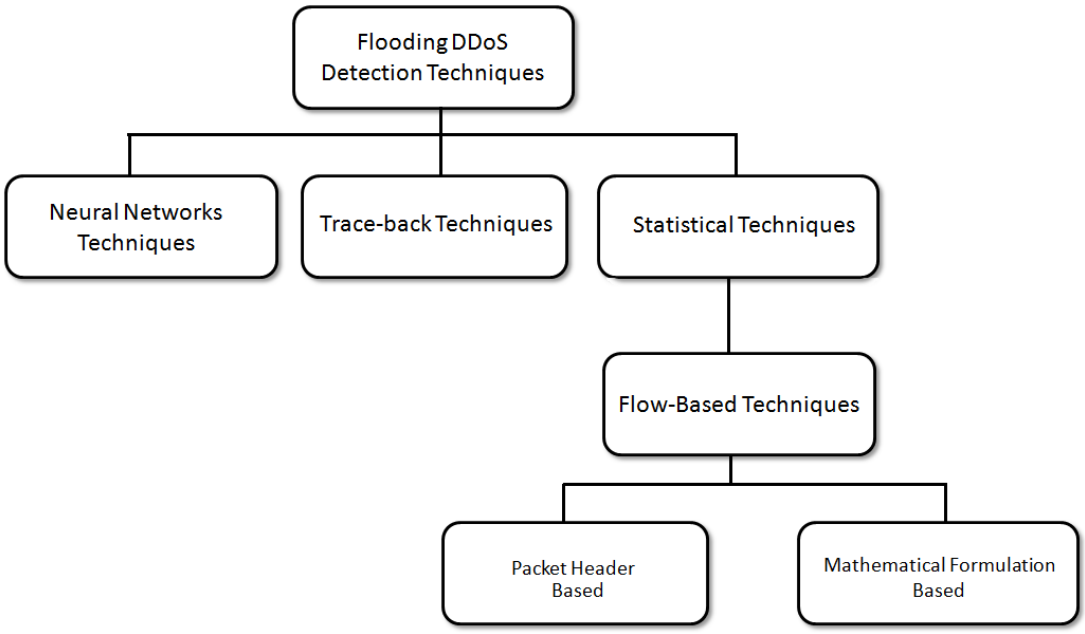

Figure 2.1: Taxonomy representing categories of solutions proposed for Flooding DDoS Detection

Linear Vector Quantization (LVQ) model of neural networks is also used to detect DDoS attacks by Li, Liu, and Gu in[24]. In [25], Karimazad and Faraahi proposed Radial Basis Function (RBF) neural networks were used to detect DDoS packets from network traffic.

An ensemble of classifiers has been used by Kumar and Selvakumaar in[26]. As a base classifier they chose Resilient Back Propagation (RBP) neural network. Various studies have proposed back propagation (BP) neural networks for detecting DDoS attacks. Agarwal and Gupta proposed back propagation scheme in [27] where various sets of traffic are fed as input and number of zombies are measured as output. Neural networks are trained with normal and attack traffics. Various traffics are fed as input and strength of each traffic with respect to DDoS attack is measured. A technique for real time estimation of DDoS attack strength and number of zombies involved is proposed in [28][29], the authors have used the BP model of neural networks for this.

2.3.1.2 Limitations of Neural Networks

The neural network schemes have many limitations. Overall, in order to understand underlying network structure sufficiently, a neural network has to be fed with large training data set. The limitation of these schemes lies in the fact that with the increase of training data, more computational and implementation cost is required, thus making these schemes inapplicable in real time large network scenarios [30].

2.3.2 Overview of Trace-back / Attacker Pinpoint Methodologies

John and Sivakumar in [83] gave survey of various trace-back schemes and explained the general characteristics that an ideal trace-back methodology should possess. They can be summarized as: It should be able to pinpoint attacker using single packet with least memory consumption and internet service providers involvement. Besides, such scheme should not reveal the identity of the tracing machine. Such schemes must be able to trace-back the attacker no matter whatsoever transformations have been applied to the attack packet.

2.3.2.1 Solutions Based on Trace-back / Attacker Pinpoint Methodologies

Lipson in [31] proposed a trace-back scheme where ICMP massage was sent with traffic, in order to know the information of the path contained in the ICMP message. This was called ICMP messaging scheme. This ICMP messaging scheme relies on the assumption that the percentage of attack packets is more than legitimate packets but this may not be the case always specially when low rate DDoS attacks are launched. Hop by hop trace-back was proposed by Kumar, Sangal and Bhandari in [32] in which the process of

attacker identification was carried out iteratively on the routers closest to the victim system towards the attack source until the attacker's source is fully traced.

Some other trace back schemes include deterministic packet marking (DPM) in which a packet belonging to a network is marked with a unique information like the first ingress edge router or sometimes the complete route. The router embeds its IP address deterministically into the IP packets. The scheme [33] was introduced to overcome some drawbacks of probabilistic packet marking (PBM) as it has simple implementation and requires less computational overhead on intermediate routers.

2.3.2.2 Limitations of Trace-back / Attacker Pinpoint Methodologies

The trace-back schemes have their own limitations. If flooding DDoS attacks consume the whole network bandwidth, the ICMP packets might be dropped thus making it difficult to trace back the attacker. In this way the whole scheme might fail. The complexity and computational cost limitations lie with hop by hop trace back methodology.

The deterministic packet marking also comes with several drawbacks. The unique information stored as a mark is only at the first edge router, reconstruction of the route requires more packets. This makes it difficult and mostly impossible to trace the true attacker source. Besides posing computational overhead with such schemes, in case of reflector attacks, the traced source IP will be of the innocent machines and not the original attacker.

While each scheme has its own limitations, some of the major ones are various assumptions that do not map onto real network scenarios, chances of false negatives and large computational power requirements [34].

2.3.3 Overview of Statistical Techniques

Statistical techniques are often applied for the detailed study of a given data. It collects and organizes data in an interpretable way. This procedure is called sampling. The next procedure that the statistical technique undergoes is data analyses, interpretation and presentation of results. Any aspect of data can be handled using statistical techniques.

2.3.3.1 Solutions Based on Statistical Techniques

In [38] and [39] a principal component analysis (PCA) techniques has been proposed. But studies indicate that PCA methodology used cannot detect anomalies effectively since inadequate methods are used to tune principal component analysis[40][41].

A stable profile maintenance idea was proposed in [42] that can detect sudden changes in network packets. Monitoring 15 packet attributed with use of relational analysis and decision trees was proposed in [43]. The metrics used were types of protocols, packet flag options, time to live and packet size.

Two statistical tests are proposed for detecting flooding DDoS Attacks. Firstly, it compares the differences involving the overall means of the incoming traffic arrival rate and the normal traffic arrival rate. If the difference is significant, it concludes that the traffic may include flooding attack packets [71].

A heuristic data structure was proposed to detect DDoS attacks called as MULTOPS [44]. The assumption was that during a normal scenario, the traffic between given two nodes is proportional. This leads to false alarms since any disproportional traffic will be detected as attack traffic which is not the case every time.

2.3.3.2 Limitations of Statistical Detection Techniques

Adjustment and fine tuning of PCA detection metrics used is a difficult task to accomplish[40][41]. MULTOPS leads to false alarms since any disproportional traffic will be detected as attack traffic which is not the case every time. Besides, the authors pointed out some failure points of MULTOPS when attack is launched from spoofed IPs since in that case, the assumption will never become true[64][65]. in [71] low rate DoS attacks cannot be detected because the tests only produce alarm when huge incoming traffic is seen. Profile maintenance idea came up with an assumption. Their assumption was that these four metrics are enough to detect instability in network traffic but the chosen metrics were not directly related to denial of service attacks and therefore, large amount of false alarms were faced in the technique [63].

2.3.4 Flow-Based Detection Techniques

We have classified the schemes into two categories as indicated in figure 2.1. i.e. Mathematical Formulation Based and Packet Header Feature Extraction Based Classification. We shall discuss about the classification in detail in Chapter 3. Following are the most significant works done in the area of flow based flooding DDoS detection:

2.3.4.1 Principal component analyses (PCA) based approaches

Principal component analyses (PCA) based approaches include studies in [48] and [49]. Network DDoS attacks were proposed to be detected by traffic decomposition to normal and abnormal divisions. The division were called the sub spaces. Studies have indicated that PCA based schemes are not practically efficient to be adopted because difficulty is faced during adjustment and fine tuning of the metrics used for attack detection [40][41][69].

2.3.4.2 D-WARD

D-WARD was proposed in [70] that acted as a linking channel between the internet and the victim network. A complete record of two way traffic, i.e. each flow record between the internet and victim network had to be kept in order to identify the attacks. The record is compared with previously stored normal network statistics. A rate limitation is applied to the identified attack traffic. Studies show that D-WARD consumes more memory space than other network based detection mechanisms [72].

2.3.4.3 Spatial & Temporal Correlation

A network wide DDoS attack detection technique was proposed in [46] in which the authors claimed to detect attacks efficiently using spatial correlation for feature extraction and temporal correlation for attack detection. This study has its own limitation because it can only detect attacks launched from spoofed IP addresses. While this is the most commonly occurring DDoS scenario, there might be real machines launching the attack with true source IP addresses [47].

2.3.4.4 Time Series Analysis Using HVM

A structural approach towards developing flow based intrusion detection system and automatic parameter tuning was proposed in [17]. A flow based time series analyses has been done for intrusion detection. For presenting the time series analysis, the authors have used Hidden Markov Models (HMMs). Unfortunately, their work has an unacceptable ratio of false positives .

2.3.4.5 IP Address Feature Value

In [59] Cheng, Yin, Liu et. al. gave a formula for IP Address Feature Value (IAFV) to detect DDoS attacks in a given flow of incoming packets. They gave a unique idea that a network flow F can be analyzed efficiently by classifying the incoming packets of the flow by source and destination IP address. The classification of packets was such that packets of one class (flow) will contain same source and destination IP addresses.

2.3.4.6 Congestion Participation Rate

In [50] flow level network traffic is used and through CPR (Congestion Participation Rate) , low rate DoS (LDDoS) attacks were proposed to be detected. Unfortunately, there scheme can only detect a small range of DoS attacks that makes it insufficient to implement in real time networks.

2.3.4.7 Profile Based NfSen Plugin

In [51] a flow based SSH dictionary attacks detection mechanism is demonstrated which they implemented as a plugin for NfSen tool. The proposed algorithm defined rules set for attacks. A profile was maintained for the incoming packets based on the rule set and

the packets were monitored in the form of flow, i.e. packets per flow per minute. The accuracy of their algorithm is yet needed to be investigated since the rules used to maintain profile need to be changed depending on the size of SSH attacks.

2.3.4.8 Flow Record Table Based Approaches

In [54] pattern of flow is recorded using flow table through which data is extracted and detection is made based upon already learnt pattern of DDoS flow like average packets per flow per unit time. However , the average will not give accurate results since some packets will have higher number of occurrences than the others. A blacklist is maintained for the detected packets , which in real scenarios is of no use since DDoS occurs from unique source IPs. Similar techniques are used in some other studies like in [56], per source IP table or a per flow table is maintained for detection. Maintaining table for each flow not only poses a scalability issue but also detecting the flows causing DDoS specially in case of flooding attacks arriving from spoofed IP packets becomes challenging.

2.3.4.9 Traffic Behavior Correlation Analyses

Using flow between attack and victim nodes, detection of DDoS attacks proactively was the technique proposed by [52]. Correlation of traffic behavior between attacker and victim machines was calculated. A normal profile is maintained in order to compare the incoming packets with that profile. The main limitation of this technique is the attack methodology and attack tool used in their scheme does not map today's complex attacks. In reality, more sophisticated attacks are encountered [53].

2.3.4.10 EWMA

In [37], the authors applied exponentially-weighted moving average (EWMA) algorithm to detect changes in incoming traffic. If the intensity of the network traffic increases with time, an alarm is raised. The main issue with such techniques is that the change point detection occurs at one time series. This might result in false alarms because in some cases flash crowd events might raise the network traffic to abnormal level for particular time[65].

2.3.4.11 Chi-Square

Assumption that during an attack the distribution of traffic is uniform gave rise to approach of [73]. They used Chi-Square statistics whose value increases during an attack. For every network and different type of attacks, the underlying baseline needs to be set for this algorithm exclusively. This process puts a burden on the detecting device and thus makes it unsuitable for real life attack scenarios.

2.3.4.12 HiFIND

Another effort done to detect flooding attacks and port scans using flow based approach was proposed in [58]. The technique was names as HiFIND (high-speed flow-level intrusion detection). They claimed to detect the attacks efficiently but studies have shown that their scheme was prone to huge false negatives and was unable to differentiate the attack events from flash events or network congestions[56].

2.3.4.13 Change Point Detection CUSUM Technique

Change point detection is a very well known and much researched technique used to detect flooding DDoS attacks. Many studies have proposed change point detection

algorithms. Some of them are improvements of the previous work done. Change point detection works on time series where the algorithm is applied to certain time series of the network traffic. A change detecting algorithm CUSUM was first pointed out in [24]. In [35] and [36], CUSUM was used to detect SYN flooding DDoS attacks. However, for detecting DDoS, a CUSUM based approach is quite complex and resource intensive. Also it requires different window size selection with respect to different datasets which makes it less applicable in real world where detection has to be done in real time [65]. In various studies, CUSUM is argued to be unsuitable for accurately detecting a DDoS attack when used alone. Also it gives high rate of false alarms as indicated in [68]. The CUSUM technique compares the incoming packets (or a particular feature of the incoming packet) with a threshold given to it. The CUSUM technique can only detect flooding events when there is a high frequency of incoming packets. This is one of the main but not the only feature that needs to be analyzed in order to mark an incoming packet as attack packet. Hence, it is insufficient to detect general DDoS attacks [66][67].

2.3.4.14 Correlation of Incoming IP Addresses

Zhongmin and Xinsheng in [55] mapped the formula of correlation coefficient on to network traffic. Their main idea was that normally the frequency of IP addresses to a destination is in a stable range but in DDoS the source IP addresses frequency becomes unstable and random since network packets belonging to random and unique IP addresses are used to attack a single target system in DDoS attack. They analyzed correlation coefficient of incoming network packets per two consecutive intervals and observed that the value of correlation coefficient is abnormally reduced during attack

conditions; since there will be larger set of unique source IP addresses per unit time. Hence, the determination of attack is based upon the values of correlation coefficient per two consecutive time periods, called as sliding window by the authors.

2.4 Summary of Flow Based Techniques

Based on reviewed literature, the existing solutions were grouped into three main categories. To date, no comprehensive solution has been proposed to detect flooding distributed denial of service attacks. Major flow-based DDoS detecting solutions along with their limitations are tabulated in Table 2.1.

2.5 Conclusion

This chapter highlights all the recent solutions proposed for flooding DDoS detection that are implemented within rule-based NIDS or exclusively along with their shortcomings. Each approach has its own limitations. There is a lack of comprehensive

Table 2.1: Summary of Flow-Based Solutions

Sr. No	Major Flow Based Proposed Scheme	Limitations
1	PCA Based Approaches[48][49]	not practical to be adopted in today's network scenario
2	D-WARD[70]	not memory efficient
3	Temporal Correlation[46]	only for spoofed attack IPs
4	Time series analyses based on HVM[17]	high false positives
5	CPR[50]	insufficient for real time implementation
6	Flow Table[54]	high false alarms, not scalable
7	EWMA [37]	cannot differentiate attack from flash events
8	Chi-Square[73]	not memory efficient
9	HiFIND[58]	high false negatives
10	Change Point Detectors [24][35][36]	high false alarms, complex, not memory efficient
11	NfSen plugin[51]	need to change profile for different attack data sets
12	Traffic Behavior Correlation Analyses[52]	insufficient for real time implementation
13	IP Address Feature Value [59]	refer to Chapter 3 and 6
14	Correlation of IP Addresses [55]	refer to Chapter 3 and 6

solution that should be adaptable to wide network range, accurate in detection, gives least false alarms and effective against today's flooding DDoS attack launching tools.

Chapter 3

Classification of Flow-Based Techniques

3.1 Introduction

In this chapter, the flow based techniques have been classified into two categories; packet header based category and mathematical formulation based category. Two flow based techniques are chosen belonging to each category. They have been chosen for their simplicity, scalability and less complexity in terms of implementation. Both of them have been explained along with their mathematical notations, algorithms and mathematical explanation.

3.2 Flow Based Detection Techniques

A new enhancement in the field of flooding DDoS detection techniques is introductions of flow based detection techniques. The idea of flow based detection is basically to analyze only a part of information from headers of incoming packets and analyze the header information by grouping the incoming packets in the form of flows. A flow is a stream of data where all packets share some or all of these characteristics: IP source and destination address, source and destination port number and protocol value[20][62]. There are many flows in a normal network traffic since many packets are coming in and getting out of the network with various source and destination IP addresses and ports respectively.

Studies indicate that flow based detection is much more scalable than a solution relying on rule based signature database[13][14][15][[21]. Lesser memory and computational resources are consumed by flow detection techniques as compared to packet based detection; since they track only header information from the incoming packets . Also they have the ability to detect novel DDoS attacks better than payload based detection mechanisms[16][17][60]. Integrating such a method with Rule Based NIDS before its detection engine will make it much more proficient.

3.3 Classification of Flow Based Solutions

To date, many flow based methodologies have been proposed. As already discussed in Chapter 2, flow based attack detection schemes are a type of statistical DDoS detection technique. We have further classified the flow based detection solutions into two categories based on the salient methodology being adopted by these in order to detect various DDoS attacks. The classification along with detection solutions and their references are given in Figure 3.1 and explained below:

Packet Header Based: The category includes the solutions in which the detection is based on various features such as source IP address, destination IP address or protocol extracted from the header of packets.

Mathematical Formulation Based: The category consists of solutions where the detection is based on mapping mathematical concepts into incoming network packets belonging to a unique flow.

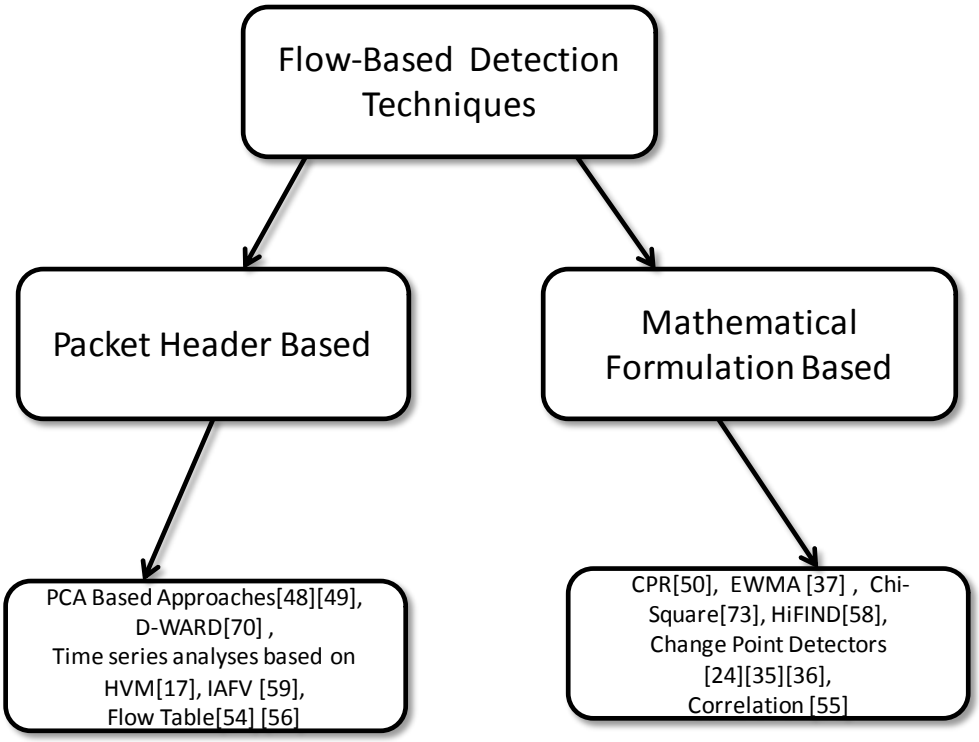

Figure 3.1 : Classification of Flow-based Flooding DDoS attack Detection Solutions

3.4 Analysis of Flow Based Solutions

It can be inferred that there is lack of a comprehensive, efficient and less complex flow based DDoS detecting solutions that have low false alarms. In search of finding such a solution, two important algorithms have been analyzed. One of the algorithm is based on IP address feature extraction [59] and belongs to packet header based category. The other chosen algorithm is based on very well known concept of probability, i.e. correlation [55] and belongs to mathematical formulation based category.

The work in [55] and [59] demonstrated that in order to detect flooding DDoS attacks, there is no need to look into complex features of incoming packet and network. Rather, simple analyses of characteristics like source IP address of incoming packets per unit

time is sufficient in order to detect that a DDoS. Their work not only minimizes the IP features that are examined but also reduce memory and CPU utilization of the system on which the algorithm will be implemented. This addresses the scalability issue of many other proposed techniques that lack this capability [24][35][36][48][49][50][54] [56][70][73].

3.4.1 IP Address Feature Value Based Algorithm

Cheng, Yin, Liu et. al. gave a formula for IP Address Feature Value (IAFV) to detect DDoS attacks in a given flow of incoming packets [59]. They gave a unique idea that a network flow can be analyzed efficiently by classifying the incoming packets of the flow by source and destination IP address. Explanation of symbols for IAFV algorithm is given in Table 3.1. The classification of packets was such that packets of one class (flow) will contain same source and destination IP addresses. SDD_i denotes number of packets arriving at i-th destination IP address. The formula for IP address feature value is as follows:

$$IAFV_f = \frac{1}{m}\sum_{i=1}^{m}(SIP(SDD_i) - m\ )$$

in which network flow is denoted as f, $SIP(SDD_i)$ is the number of different source IP addresses in the class SDD_i and m is the total number of destination IP addresses.

Let the threshold for attack be denoted as ST such that:

(i) If $IAFV_f >$ ST, then it is declared that attack has occurred.

(ii) If $IAFV_f \leq$ ST, then it is declared that no attack has occurred.

3.4.1.1 False Negatives in IAFV

The IAFV value is the subtraction of number of destinations from the summation of all incoming packets with unique source IP addresses from all the destinations. A situation can occur if flooding DDoS is launched on single destination, e.g. destination x , but the destination *y* and *z* are not under attack, and the overall IAFV value is such that such that the threshold limit is not exceeded, then no attack alarm will be triggered for any of the destinations. This situation will lead to false negatives since it is unable to distinguish between the destination under attack. Figure 3.2 explains the IAFV algorithm.

Let "*M*" be a unique set of destinations such that:

$$|M| = m$$

where $\{x, y, z\} \in M$

Let "*A*" be a set of all unique source IP addresses:

$$A = \{ SIP(SDD_X), S\ IP(SDD_Y), SIP(SDDz)\}$$

Let $ST_i = ((SIP(SDD_i) - m)$

And $(\exists x \in M): ST_X > ST\ )$

$(\exists y \in M): ST_Y \leq ST\ )$

$(\exists z \in M): ST_Z \leq ST\ )$

Such that: $\frac{1}{m}\sum_{i=1}^{m} SIP(SDD_i) - m \leq ST$

In this case, since IAFV $\leq$ ST, no attack will be detected whereas, destination x $\in M$ is under attack as discussed.

Table 3.1: Symbols For IAFV Algorithm

Symbols	Meanings
SDD_i	number of packets arriving at ith destination IP address
$SIP(SDD_i)$	number of different source IP addresses at ith destination IP address
m	total number of destinations
ST_t	threshold for DDoS attack detection

```
Input: Packets of Flow F, a sample interval Δt, a criteria to
stop algorithm

Q,a source IP address S, a destination IP address D, an IP
address class set SD, SDS and SDD, an IP address features IAFV.

Output: IAFV  results.

1:      Initialize the variables;

2:     While (Q is not fulfilled)

3:           Read the T, S and D of an IP packet from F;

4:     End While

5:     if (time exceeds the decided Δt)

6:           Add all packets with different destination and same
source in SDD

7.           Count m \\m is the number of the elements in SDD.

8:     End if

9:     calculate IAFV

10:    return IAFV
```

Figure 3.2 : IAFV Algorithm

3.4.1.2 False Positives in IAFV

A situation may arrive where flooding DDoS is launched on single destination, e.g. destination *x*, but the destination *y* and *z* are not under attack, still the overall IAFV value is such that such that the threshold limit is exceeded from threshold, then the attack alarm will be triggered for all the destinations.

Let "*M*" be a unique set of destinations such that:

$|M| = m$

where $\{x, y, z\} \in M$

Let "*A*" be a set of all unique source IP addresses:

$A = \{ SIP(SDD_x), S\ IP(SDD_y), SIP(SDDz)\}$

Let $ST_i = ((SIP(SDD_i) - m)$

And $(\exists x \in M): ST_x > ST\ )$

$(\exists y \in M): ST_y \leq ST\ )$

$(\exists z \in M): ST_z \leq ST\ )$

Such that: $\frac{1}{m}\sum_{i=1}^{m} SIP(SDD_i) - m \geq ST$

In this case, since IAFV $\geq$ ST, attack will be detected at all destinations whereas, destination y and z $\in M$ are not attack as discussed.

3.4.2 Correlation Algorithm

Zhongmin and Xinsheng mapped the very well known and acknowledged concept of probability that is correlation coefficient on to network traffic. For details about correlation coefficient please refer to [13] and [14]. Their main idea was that normally the frequency of IP addresses to a destination is in a stable range but in DDoS the source IP addresses frequency becomes unstable and random since network packets belonging to random and unique IP addresses are used to attack a single target system in DDoS attack.

They analyzed correlation coefficient of incoming network packets per two consecutive intervals and observed that range of value of correlation coefficient is steady normally while during attack circumstances, the value of correlation coefficient is abnormally reduced since there will be a larger set of unique source IP addresses per unit time. Hence, the determination of attack relies upon the values of correlation coefficient per two consecutive time periods, called as sliding window by the authors. Table 3.2 gives explanation of the symbols used in the algorithm.

3.4.2.1 False Negatives in Correlation Algorithm

It is possible that the incoming packets belong to attack packets, such that the correlation coefficient value does not exceed than the threshold, then the attack will not be detected. This situation will also lead to false negatives. The results verify this proof and can be seen in Chapter 6. The following expression will show a condition that leads to false negative results.

Let the threshold for attack be denoted as ρt.

Let "*A*" be a set of IP addresses with attack packets.

$(\exists x(n) \in A) \mid \rho \leq \rho t$)----------(11)

3.4.2.2 False Positives in Correlation Algorithm

If a sudden burst of normal incoming packets arrive that actually do not belong to attack packets, such that the correlation coefficient value exceeds then the threshold, then the traffic will be detected as attack traffic.

Table 3.2: Symbols For Correlation Algorithm

Symbols/Formulas	***Meanings***
M	Number of IP addresses in a sliding window time interval
k	Refers to k-th time interval
n	n-th sliding window time interval
$t_{1,2}$	Refers to the sliding window time interval between given 1st and 2nd second
$X_i(n) = \sum_{k=1}^{3} X_i(n_k)$	Data sent by the i-th packet in n-th sliding window
$X(n) = \sum_{i=1}^{M} X_i(n)$	The total amount of data packets sent by all IP addresses in the n-th slide window time interval
E(x(n)) = $\sum_{i=1}^{M} X_i(n) \frac{X_i(n)}{X(n)}$	Mathematical expectation of the number of data packets of every IP address in a slide window time interval
$\rho = \frac{\sum_{i=1}^{M}(x_i(n) - E(x(n)))((x_i(n+1) - E(x(n+1)))}{\sqrt{\sum_{i=i}^{M}(x_i(n) - E(x(n)))}\sqrt{\sum_{i=1}^{M}(x_i(n+1) - E(x(n+1)))}}$	Correlation Coefficient Formula

The results verify this proof and can be seen in Chapter 6. Figure 3 gives the correlation algorithm.

Let "*B*" be a set of IP addresses with non-attack packets.

($\exists x(n) \in B$) | $\rho \geq \rho t$)----------(12)

Input: *A network data in form of flow F, a sliding window time interval* $t_{1,2}$, *source IP addresses xi(n), correlation coefficients* ρ

Output: *Correlation coefficient values*

1: **For** *(sliding window time interval* $t_{1,2}$*)*

2: *Calculate xi(n) \\ xi(n) is the number of packets of a unique source IP address*

3: **End For**

4: ***For****(all the packets)*

5: *Calculate x(n) \\ Calculate the amount of all data packets in that window time interval*

6: **End For**

7: **For** *(each data packet in sliding window time interval* $t_{1,2}$*)*

8: *Calculate E(x(n)) and E(x(n+1) \\ Calculate the mathematical expectation of each IP data packets in that sliding window time interval*

9: **End For**

10: **For** ***($t_{1,2}$)***

11: *Calculate* $s=s+(xi(n)-E(x(n)))*(xi(n+1)-E(x(n+1)))$

12: *Calculate* $dx(n)=dx(n)+(xi(n)-E(x(n)))^2$

13: *Calculate* $dx(n+1)=dx(n+1)+(xi(n+1)-E(x(n+1)))^2$

14: *Calculate* $\rho = \frac{s}{\sqrt{dx(n)}\sqrt{dx(n+1)}}$ *\\ correlation coefficient for* $t_{1,2}$

15: **End For**

16: **End For**

Figure 3.3 : Correlation Algorithm

3.5 Conclusion

In this chapter, main focus has been given on the two chosen flow based flooding DDoS detection solutions. Both of the algorithms have been explained in detail. The reason behind choosing them is that most of the flow based techniques proposed to detect flooding DDoS attacks pose scalability issue and are not memory efficient besides having high rate of false alarms. From mathematical explanation of both algorithms in Section 3.4.1 and 3.4.2, it has been found that in certain scenarios, there are chances of false positives and false negatives in both schemes.

Chapter 4

Proposed Solution

4.1 Introduction

This chapter attempts to provide a solution to the flooding DDoS attack detection problem, and outlines the design of an extended flooding DDoS detection strategy. Flooding distributed denial of service attacks first hit the network almost more than a decade ago [45] where a set of compromised nodes/machines were commanded by their master (main attacker) to launch high volume of legitimate but unwanted traffic towards the victim machine. Within the internet community, the flooding DDoS attack detection continues to represent a very hazardous threat as indicated in [81]. The problem with the detection of flooding DDoS attack is that the requests sent by the compromised bots or the tools used for launching attack are legitimate and hence, it is a challenging problem to differentiate between a legitimate request and an attack request. In this Chapter, an improved design of flooding DDoS attack detection technique based on an existing correlation technique [55] is presented. The existing correlation technique is based on of the change in rate of new source IP addresses of the incoming packets per two consecutive time intervals while the improved correlation technique is based on of the change in rate of new source IP addresses of the incoming packets over multiple sliding window time intervals.

4.2 Issues in the Existing Correlation Algorithm

A large number of unsolicited packets originating from unseen or random source IP addresses is the main indicator of the onset of a flooding DDoS attack. This was the characteristic feature chosen by Correlation algorithm [55].

The Correlation algorithm used a single sliding window time interval as a time scale to analyze the network flow. Although this algorithm solves the scalability issue; since only feature needed to be extracted in order to be fed into the system is source IP address of the packet, the Correlation technique is prone to false positives and false negatives.

If a burst of non attack traffic, called flash crowd arrives in that time interval, it raises the alarm and produces false positives. Also, if for that certain periods of time, the correlation between attack packets is higher than the threshold, no attack will be detected and thus produces false negatives.

4.3 Proposed Correlation Algorithm (MSW-Correlation)

The proposed technique has been named as Multiple Sliding Window Correlation Technique (MSW-Correlation). It extends the previous work done in this direction by [55]. They analyzed correlation coefficient of incoming network packets per two consecutive intervals and observed that the value of correlation coefficient is abnormally reduced during attack conditions; since there will be larger set of unique source IP addresses per unit time. Hence, the determination of attack is based upon the values of correlation coefficient per two consecutive time periods, called as sliding window. An enhancement has been introduced in this technique, where sliding window time interval correlation

analyses has been calculated using correlations of 4 consecutive sliding windows, in order to reliably determine if the current incoming network traffic represents attack condition or not.

4.3.1 Notations Used for MSW-Correlation Technique

While other notations are the same as described in Chapter 3, the newly introduced notations are defined in Table no. 4.1.

4.3.2 Steps for MSW-Correlation Technique

The modified algorithm is explained using a flowchart in figure 4.3. Each of the step is explained as follows:

4.3.2.1 Multiple Sliding Window Time Intervals

As already discussed, in order to reliably determine if the current incoming network traffic represents attack condition or not, sliding window time interval correlation analyses has been calculated using correlations of 4 consecutive sliding windows. Figure 4.1. illustrates time slicing of multiple sliding window time intervals that have been used in the proposed correlation algorithm along with correlation notations for each sliding window time interval.

Let *x* and *y* be any two consecutive time instants such tha y > x.

Then: $t_{x,y}$ denotes *x-th* and *y-th* time interval between instants x and y, where y > x And: ρi denotes *i-th* correlation coefficient

Table 4.1: Symbols for MSW-Correlation Algorithm

Terminology	*Symbols/Formulas*	*Meaning*
Sliding window time intervals	$t_{1,2}$	Time interval between first and second time instants
	$t_{1,3}$	Time interval between first and thrid time instants
	$t_{1,4}$	Time interval between first and fourth time instants
	$t_{1,5}$	Time interval between first and fifth time instants
Mathematical expectation of the number of data packets of every IP address in n-sliding window time interval	$E(x(n))$	Expectation value of packets in first time instant
	$E(x(n+1))$	Expectation value of packets in 2nd time instant
	$E(x(n+2))$	Expectation value of packets in 3rd time instant
	$E(x(n+3))$	Expectation value of packets in 4rth time instant
	$E(x(n+4)$	Expectation value of packets in 5th time instant
Correlation Coefficient Formula	$\rho_1 = \frac{\sum_{i=1}^{M}(x_i(n)-E(x(n)))(x_i(n+1)-E(x(n+1)))}{\sqrt{\sum_{i=1}^{M}(x_i(n)-E(x(n)))^2}\sqrt{\sum_{i=1}^{M}(x_i(n+1)-E(x(n+1)))^2}}$	Correlation Coefficient of First and 2nd time instants
	$\rho_2 = \frac{\sum_{i=1}^{M}(x_i(n)-E(x(n)))(x_i(n+2)-E(x(n+2)))}{\sqrt{\sum_{i=1}^{M}(x_i(n)-E(x(n)))^2}\sqrt{\sum_{i=1}^{M}(x_i(n+2)-E(x(n+2)))^2}}$	Correlation Coefficient of First and 3rd time instants
	$\rho_3 = \frac{\sum_{i=1}^{M}(x_i(n)-E(x(n)))(x_i(n+3)-E(x(n+3)))}{\sqrt{\sum_{i=1}^{M}(x_i(n)-E(x(n)))^2}\sqrt{\sum_{i=1}^{M}(x_i(n+3)-E(x(n+3)))^2}}$	Correlation Coefficient of First and 4rth time instants
	$\rho_4 = \frac{\sum_{i=1}^{M}(x_i(n)-E(x(n)))(x_i(n+4)-E(x(n+4)))}{\sqrt{\sum_{i=1}^{M}(x_i(n)-E(x(n)))^2}\sqrt{\sum_{i=1}^{M}(x_i(n+4)-E(x(n+4)))^2}}$	Correlation Coefficient of First and 5th time instants

4.3.2.2 Packet Count for each Sliding Window Time Interval

For each sliding window time interval $t_{1,2}$, $t_{1,3}$, $t_{1,4}$ and $t_{1,5}$, the total number of packets coming from all source IP addresses is calculated. This will give clear statistics of how many source IP address have been received by victim in each sliding window interval. Using this data along with the unique source IP addresses count for each sliding window time interval, the expectation values for each sliding window time interval is calculated and fed into correlation coefficient formula.

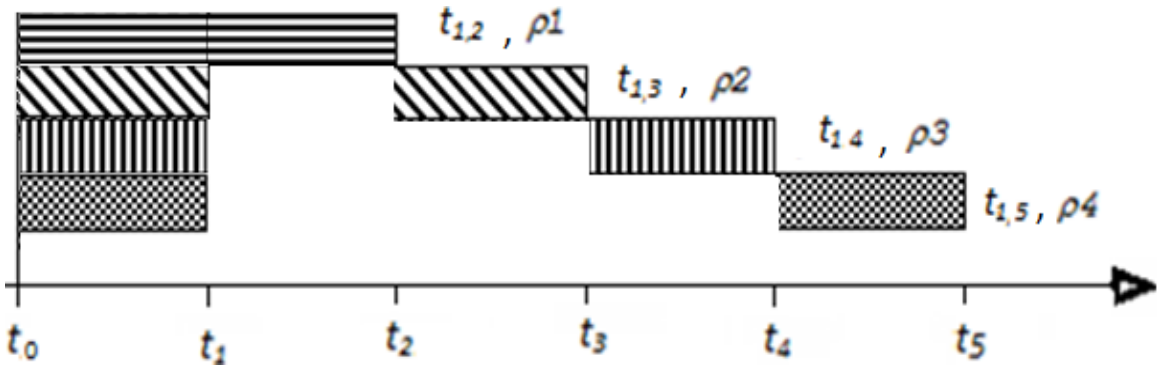

Figure 4.1: Time Slicing for MSW-Correlation Algorithm

The correlation algorithm is as follows:

Input: A network data in form of flow F, 4 sliding window time intervals $t_{1,2}$, $t_{2,3}$, $t_{3,4}$ *and* $t_{4,5}$, *source IP addresses xi(n), correlation coefficients* $\rho 1$, $\rho 2$, $\rho 3$, $\rho 4$ *of each sliding windows time interval* $t_{1,2}$, $t_{1,3}$, $t_{1,4}$ *and* $t_{1,5}$.

Output: *Correlation coefficient values*

1: **For** *(each sliding window time interval* $t_{1,2}$, $t_{1,3}$, $t_{1,4}$ *and* $t_{1,5}$*)*

2: *Calculate xi(n) \\ xi(n) is the number of packets of a unique source IP address in that particular slide window time interval*

3: **End For**

4: ***For****(all the packets of a particular sliding window time interval)*

5: *Calculate x(n) \\ Calculate the amount of all data packets in a slide window time interval*

6: **End For**

7: **For** *(each data packet in each sliding window time interval* $t_{1,2}$, $t_{1,3}$, $t_{1,4}$ *and* $t_{1,5}$.*)*

8: **For** *(a=0 to 4)*

9: *Calculate E(x(n+a));\\ Calculate the mathematical expectation of each IP data packets in each time interval* $t_{1,2}$, $t_{1,3}$, $t_{1,4}$ *and* $t_{1,5}$.

10: **End For**

11: **End For**

12: **For** *(each sliding window time interval* $t_{1,2}$, $t_{1,3}$, $t_{1,4}$ *and* $t_{1,5}$.*)*

13: **For** *(b=1 to 3)*

14: *Calculate* $s=s+(xi(n)-E(x(n)))*(xi(n+b)-E(x(n+b)))$

15: *Calculate* $dx(n)=dx(n)+(xi(n)-E(x(n)))^2$

16: *Calculate* $dx(n+1)=dx(n+b)+(xi(n+b)-E(x(n+b)))^2$ *//Calculate the values of nominator (s) and denominator (d)*

17: *Calculate* $\rho 1$, $\rho 2$, $\rho 3$, $\rho 4$ *\\ correlation coefficient for four adjacent sliding window time intervals*

18: **End For**

Figure 4.2: MSW-Correlation Algorithm

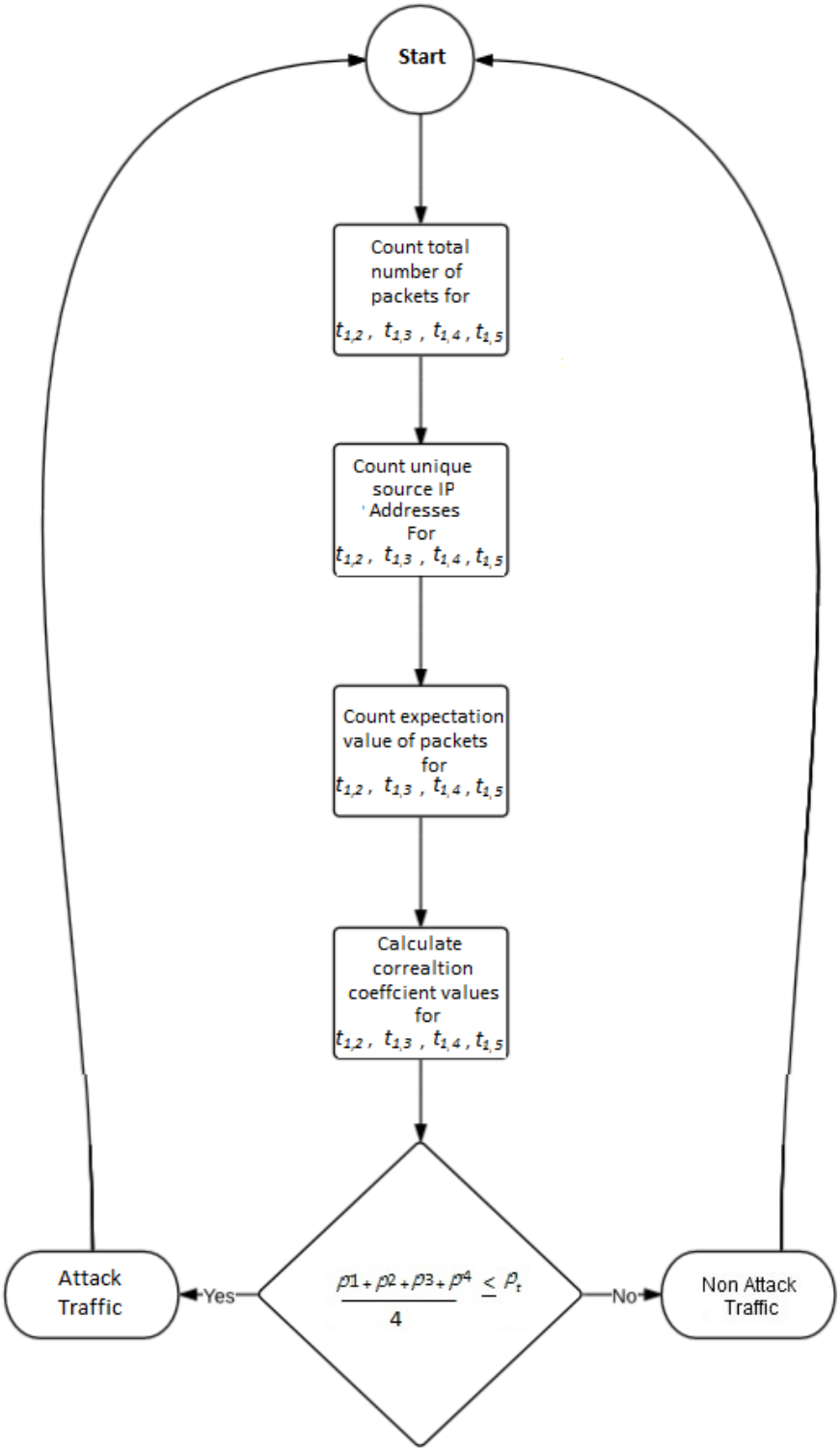

Figure 4.3: Flow Chart of MSW-Correlation Algorithm

4.3.2.3 Unique Source IP Address Count for Each Sliding Window Time Interval

For each sliding window time interval $t_{1,2}$, $t_{1,3}$, $t_{1,4}$ and $t_{1,5}$, the number of packets coming from unique source IP address is calculated. This will give clear statistics of how many unique source IP address have been received by victim in each sliding window interval. Using this data along with the total packet count for each sliding window time interval, the expectation values for each sliding window time interval is calculated and fed into correlation coefficient formula.

4.3.2.4 Average of Correlation Analyses Over Multiple Time series

As indicated before, the previously discussed correlation algorithm has used a single sliding window time interval as a time scale to analyze the network flow, hence, if a burst of non attack traffic arrives in that time interval, it considers that an attack traffic. Also sometimes, in a single sliding window time interval, attack traffic does not make a noticeable change in the correlation coefficient value.

Hence, another modification has been done in the algorithm. It is based on the observation that in order to correctly determine the presence or absence of an attack, multiple sliding window time intervals must be taken. Attack monitoring is done with average of every calculated $\rho 1$, $\rho 2$ $\rho 3$ and $\rho 4$. The decision that an incoming traffic is attack is made only if the average value of $\rho 1$, $\rho 2$, $\rho 3$ and $\rho 4$ is less than or equal to the threshold ρt:

$$\frac{\rho 1 + \rho 2 + \rho 3 + \rho 4}{4} \leq \rho t$$

4.3.3 Flow-Chart for MSW-Correlation Technique

The flow chart explains the main procedures of the proposed correlation algorithm. The total number of packets in each sliding window time interval is calculated, then the unique source IP addresses in each sliding window time interval is calculated. Correlation coefficient value is calculated for each slinging window time interval. The decision is made based on the average value of $\rho 1$, $\rho 2$, $\rho 3$ and $\rho 4$

4.4 Conclusion

Most of the techniques deployed are insufficient to detect flooding DDoS attacks either due to either scalability issue or structural weakness or lack of accurate detection that leads to false alarms as discussed in Table 2.1 of Chapter 2. In this chapter, a correlation based flooding DDoS detection technique is proposed, which is an extension of the work done in [55] and it aims to limit the false positives and false negatives. It does so by making use of multiple sliding window time intervals analyses to improve the identification of malicious traffic.

Chapter 5

Implementation and Testing

5.1 Introduction

Since the main goal of this thesis is to introduce a better flooding DDoS detecting mechanism in to rule-based network intrusion detection system, the network intrusion detection system Snort has been chosen as the subject NIDS as it has achieved the position of de-facto standard among all the NIDS. It is a rule-based NIDS that depends mainly on its signature database for attack detection and is famous for being open source, lightweight and producing expected results when an attack packet matches with any of the rules present in its database [18]. Two test benches have been used, one comprising of physical systems called Test-bed 1 while the other is emulation based on DeterLab called Test-bed 2.

5.2 Snort Architecture

Figure 5.1 shows the system architecture of Snort. In attack detection mode, various modules are used to read attack signatures and match them with the incoming traffic, if the traffic is found to be legitimate then is passed on to next module else an alert is generated and the event is recorded/logged if a rule exists. The main flow of various modules of Snort is described in the following sections.

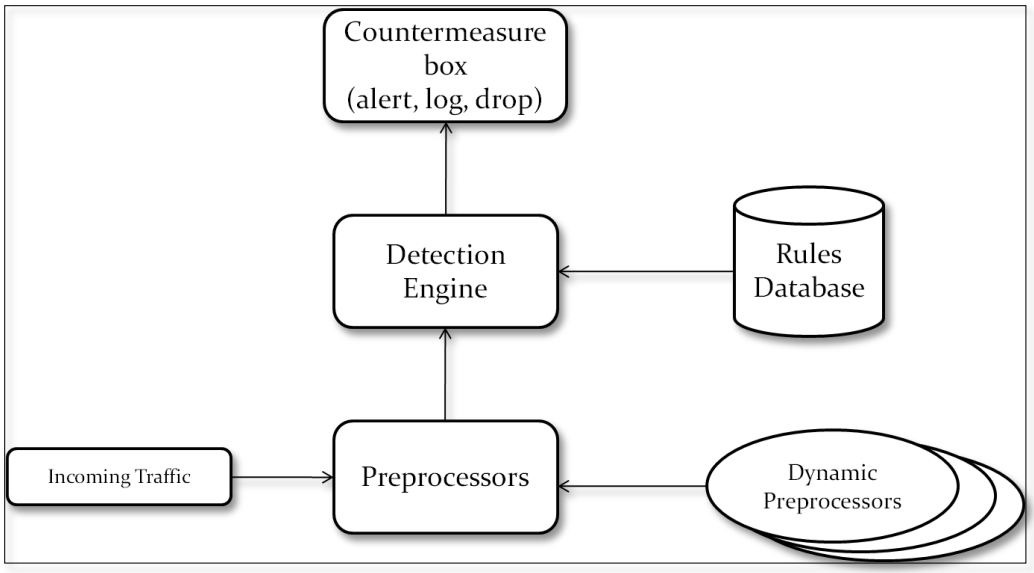

Figure 5.1: Snort Overview

5.2.1 Packet Decoder:

This module has the responsibility to observe the protocols of the incoming raw packets from all TCP/IP layers. All the information about all packets is stored in the form of a data structure. Next modules make use of this data structure for their processing.

5.2.2 Preprocessors:

From decoder, the packets are then sent to preprocessors. There are numerous preprocessors in Snort, each working for a certain attack identification. The order of preprocessors through which each packet will be checked can be prioritized and changes from Snort configuration file. Main tasks performed by preprocessors are packet fragmentation, normalization and stream reassembly. if the traffic is found to be legitimate then is passed on to detection engine else an alert is generated and the event is recorded/logged if a rule exists.

5.2.3 Detection Engine:

This is the module where actual attack identification and detection is carried out. This is done by matching each incoming packet with signature database which has been formulated on the basis of previously defined and stored rules. If an attack is suspected, the packet is either dropped or passed depending upon the applicable rules.

5.2.4 Logging and Alerting System:

This modules takes information from the detection engines and either logs packet(s) or generates alerts if attack traffic is found or both.

5.2.5 Snort Version Installed

In this thesis, the version of Snort used was 2.9.6.0 installed from [75]. This was found to be the latest stable version at the start of experimentation period. No significant changes have been introduced in the newer versions of Snort in terms of detecting flooding DDoS attacks.

5.2.6 Integration of Proposed Algorithm With Rule Based Network Intrusion Detection System:

The main issue with Snort is that it lacks the ability to detect attacks that do not match with any of the signatures. As flooding DDoS attacks do not match with any signature in particular, they remain undetected. A Rate_filter feature has been added to Snort that aims at filtering packet based on the number of incoming packets from a particular source or to a particular destination per unit time and was introduced to prevent DDOS attacks in Snort. For the purpose of detecting flooding DDOS attacks and various port scans, Snort developers' team added a feature of rate_filter with 2.8.5 version. It limits

incoming packets by either source or destination IP address based on number of packets received by the machine per second. Limitation is done by dropping further incoming packets for a unit of time decided by network administrator. This feature is not very effective and often leads to false negatives.

Both chosen algorithms [55] and [59], as well as the proposed algorithms (refer to Chapter 4 for details) have been integrated with Snort as its dynamic preprocessor. More detail about dynamic preprocessors of Snort and the methodology of integration is given in section 5.3.

5.2.7 Dynamic Preprocessor for Snort

It is possible to develop dynamically loadable preprocessors for Snort, which can be run outside Snort while using dynamic libraries and certain functions of the Snort source code. A dynamic preprocessor module for Snort has been developed based upon the proposed correlation technique.

Snort performs many pre-operations before the network packet is sent to signature database inside the detection engine. These pre-operations are accomplished by preprocessors. Preprocessors are able to perform complex analysis on packets which are otherwise, not possible to do inside rule-based detection engine. There are many preprocessors in Snort but as already discussed, are insufficient to detect flooding DDoS attacks. Major header file that need to be understood and possibly edited to develop the module are:

1- SFSnortPacket data structure is the main source of information contained inside an incoming packet. It is inside a header file with Snort source code sf_snort_packet.h. This header file contains the current data structures of a given network packet. It is major header file that is used for development of dynamic preprocessor module of Snort.

2- DynamicPreprocessor is another important data structure that is used to develop dynamic preprocessor module of Snort. It registers the preprocessor, makes it able to start, exit, restart and execute the main processing function. It has the functions for logging, exceptions, fatal errors and debugging information etc. It is defined in the header file sf_dynamic_preprocessor.h.

3- Another important header file that is needed for development of dynamic preprocessor module of Snort is sf_packet_info.h. It contains information like preprocessor name, version and main packet processing function of the preprocessor.

5.3 Traffic Generation Tools

This section explains the tools that have been used to generate and deploy flooding DDoS attack traffic and normal traffic which closely resemble real-world scenarios. It is worth mentioning that there is a strong lack of attacks representing current and novel DDoS scenarios in the old data sets as DARPA or KDD Cup 1999 Dataset [78][79][80].

A realistic traffic generation framework has been co-operatively developed as a part of this research in order to synthetically generate and deploy different attack and normal traffic scenarios which closely resemble real-world scenarios. The framework makes use of modest hardware and exploits the random IP generation feature of various attack generating tools, that is used for sending network packets with multiple distinct IP

addresses from a single source machine to a single destination machine. Thus, synthetically generating normal and anomalous traffic originating from a single machine that is able to generate wide range of source IPs using the packet generating tools. The tools used for background and attack traffic generation are mentioned in Table 5.1

5.4 Network Architecture For Attack Scenarios

For both test-bed 1 and 2, figure 5.3 shows the basic network architecture for the experimental setup of attacks. All the machines are connected using a manageable high performance switch. The switch also mirrors the victim traffic towards Snort machine through SPAN feature so that Snort is able to receive every network packet coming into or going to the victim machine.

5.5 Network Architecture for Normal Traffic

For both test-bed 1 and 2, figure 5.4 shows the basic network architecture for this experimental setup of normal traffic. Apache Web Server is set up at victim system. Instead of attacker machine, a large number of legitimate TCP Syn packets are sent towards the victim web server . Like in attack traffic scenario, the switch connects all the four machines. The switch also mirrors the victim traffic towards Snort machine through SPAN feature so that Snort is able to receive every network packet coming into or going to the victim machine.

5.6 Normal Traffic Test Scenarios

Just like attack scenarios which differ in randomness of incoming source IP addresses of the packets, both old and proposed technique have been tested for the degree of false

Table 5.1: Tools Details

Machines	Tools
Rule-Based NIDS	Snort Version 2.9.6.0 [75]
Background Traffic Generating Machine	TcpReplay Version 2.3.5 [86] ,Ostinato Version 0.5.1 [88]
Attacking Machine	Hping3 [87] , Ostinato Version 0.5.1
Victim Machine	Apache Web Server 2.4.10, Team viewer 9, Wireshark 1.12.0 [89]

positives they produce under different number of unique source IP addresses. Since IAFV algorithm is claimed to be applied on multiple destinations, the test scenarios for IAFV algorithms have been explained in separate table, Table 5.4. The total traffic has been distributed among 10 destinations with different percentage of attack and normal traffic sent to different destinations. Total 9 test scenarios have been created and tested. The test scenarios for normal traffic are given in Table 5.2 and 5.4 for Snort, Correlation Algorithm and IAFV algorithm respectively.

5.7 Attack Traffic Test Scenarios

Snort drops packets after a certain limit, mainly due to hardware limitations. Therefore, this thesis considers experimenting within the limit of packet rate after which Snort begins to drop packets. Therefore, the first step in the evaluation procedure was to determine the legitimate rate of incoming traffic that could be handled by Snort [13] [74].

The detection capability of both old and proposed technique have been tested under different degree of uniqueness of source IP addresses of the incoming attack packets.

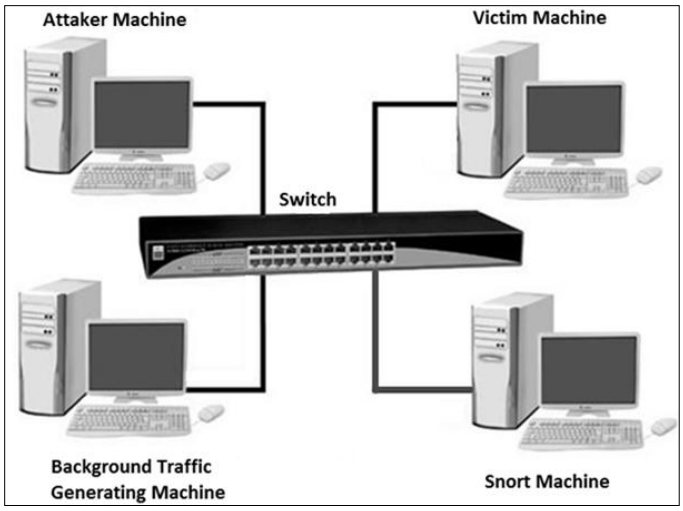

Figure 5.3 Network Architecture For Attack Traffic Test Scenarios

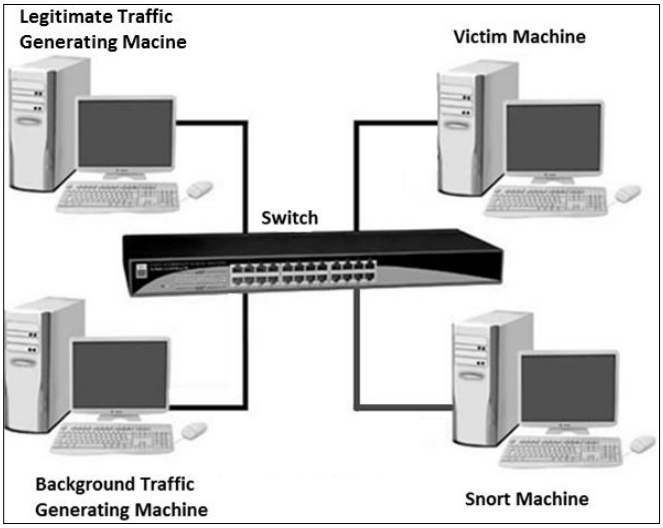

Figure 5.4: Network Architecture For Normal Traffic Test Scenarios

Since IAFV algorithm is claimed to be applied on multiple destinations, the test scenarios for IAFV algorithms have been explained in separate table, Table 5.4. The total traffic has been distributed among 10 destinations with different percentage of attack and normal traffic sent to different destinations. Total 9 test scenarios have been created

and tested. The attack scenarios can also be seen in Table 5.3 and 5.4 for Snort, Correlation Algorithm, MSW-Correlation Algorithm and IAFV Algorithm respectively.

5.8 Test-Bed 1: Design Using Real Systems

The proposed experimental framework makes use of modest hardware. NIDS show limited performance when running on virtual platforms as indicated in [76], therefore, the test bench comprises of real environment, each including four systems for conducting the experiments. It has been shown in [77] that Linux is a better operating system as compared to windows operating system in terms of Snort implementation hence, Linux has been installed in all systems.The first test-bench has been carefully designed according to the requirements of the network under test. Linux operating system has been installed on each computer system. The test bench design is very simple and it enables to convenient management of systems and construction of exclusive flooding DDoS testing facility. The implemented test-bed has been described in Table 5.5. The operating system used for each machine is Ubuntu 11.04. The experimentation has been carried out in a real environment, which ensures the best accuracy.

5.9 Test-Bed 2: Emulation Using DeterLab

DETERlab gives researchers the opportunity to conduct repeatable medium-scale Internet emulation experiments for a broad range of network security projects. For predictable results, realistic large scaled resources play an important role[57]. DETERlab is a world class state-of-the-art computing facility by USC/ISI and its abbreviation of

"cyber Defense Technology Experimental Research Laboratory". Table 5.6 gives the systems details used in the DETERlab experiments.

Table 5.2: Normal Traffic Test Scenarios for Snort , Correlation and MSW-Correlation Algorithm

Sc. No.	Increase in Unique Source IP Addresses	For Test-Bed 1	For Test-Bed 2
		Unique Sources (Packet per second)	Unique Sources (Packet per second)
N-1	20%	1000	240
N-2	50%	2500	600
N-3	70%	3500	840

Table 5.3: Attack Test Scenarios for Snort , Correlation and MSW-Correlation Algorithm

Sc. No	Unique Attack Traffic (%)	For Test-Bed 1		For Test-Bed 2	
		Attack Packets (Packet per second)	Normal Packets (Packets per second)	Attack Packets (Packet per second)	Normal Packets (Packets per second)
A-1	20	1000 (each from unique source)	4000, 52 IPs each sending 75	240 (each from unique source)	960, 34 IPs each sending 28
A-2	30	1500(each from unique source)	3500, 46 IPs each sending 75	360(each from unique source)	840, 30 IPs each sending 28
A-3	40	2000(each from unique source)	3000, 39 IPs each sending 75	480(each from unique source)	720, 25 IPs each sending 28
A-4	50	2500(each from unique source)	2500, 33 IPs each sending 75	600(each from unique source)	600, 21 IPs each sending 28
A-5	60	3000(each from unique source)	2000, 26 IPs each sending 75	720(each from unique source)	480, 17 IPs each sending 28
A-6	70	3500(each from unique source)	1500, 20 IPs, each sending 75	840(each from unique source)	360, 13 IPs each sending 28

The script tells which operating systems are to be used and which software needs to be installed on each node with boot and link speeds. Each experiment has its own file system that can be mounted from the user's experimental nodes.

Table 5.4: Test Scenarios for IAFV Algorithm

Scenario No.	Percentage of Normal Traffic	Percentage of Attack Traffic	Number of Destinations Receiving Normal Traffic	Number of Destinations Receiving Attack Traffic
1	80	20	5	5
2	80	20	9	1
3	80	20	1	9
4	50	50	5	5
5	50	50	9	1
6	50	50	1	9
7	20	80	5	5
8	20	80	9	1
9	20	80	1	9

Table 5.5: Systems Details for Test-Bed 1

Machines	Operating Systems	CPU	Memory
Rule-Based NIDS	Ubuntu 11.04	Intel Core i5, 3.20 GHZ, 1600 MHz	4 Gb
Background Traffic Generating Machine			2 Gb
Attacking Machine	BackTrack R3		4 Gb
Victim Machine	Ubuntu 11.04		4 Gb
Switch	Cisco Catalyst 2960 Switch (48) 10/100 Ethernet ports [90]		

Each registered user can access his own experimental nodes using SSH. A programmable backplane of Ethernet provides network connection to the experimental nodes. Each node is connected to that switch through VLANs, which are used to create desired network topology for the respective experiments. Besides, each node has at least a 100Mbs port for downloading various software and controlling the experiment. No node has any connection with external internet.

The *DETERlab* test-bed uses the *Emulab* cluster test-bed software developed by the University of Utah [85]**.** A researcher can use systems from a variety of computer

systems available. To date, more than 3000 experiments have been conducted using emulation of DETERlab. **The test bed provided by DETERlab is used by h**undreds of institutions worldwide. The projects based on DETERlab include network behavior analyses, worm detection, DDoS attacks detection and launching, encryption and pattern detection. For guidance, full documentation including DETERlab usage policy and many tutorials on setting DETERlab nodes and NS files has been provided to its users.

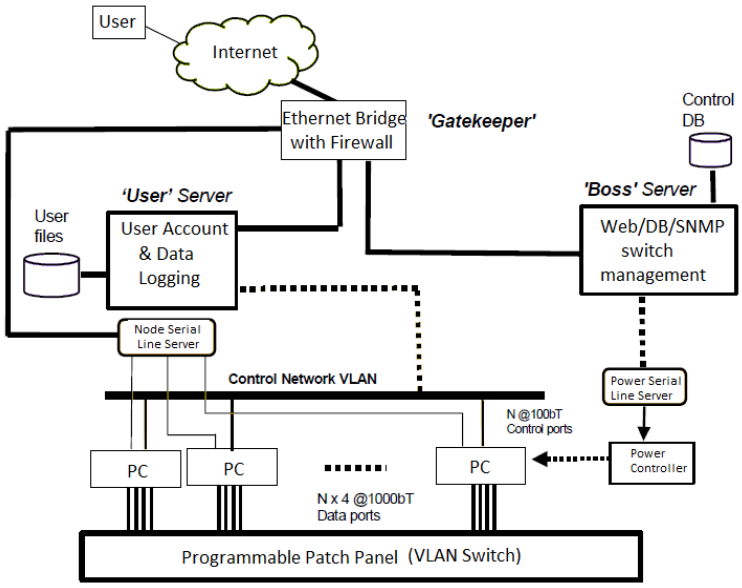

Figure 5.5 : Schematic of Deter Test-Bed [from 84]

Table 5.6 Systems Details for Test-Bed 2

Machines	**Operating Systems**	**CPU**	**Memory**
Rule-Based NIDS	Kali Linux 1.04	Intel (R) Xeon (TM) CPU 3GHz, 3600 MHz	2 Gb
Background Traffic Generating Machine			2 Gb
Attacking Machine	Kali Linux 1.04		2 Gb
Victim Machine	Kali Linux 1.04		2 Gb

Documentation of DETERLab also includes information on available nodes, hardware on the nodes, operating system images that can be loaded on the nodes, how to customize the operating systems and how to install or transfer data or software to the nodes.

5.10 Threshold:

To ensure accurate and real time detection, network data from MIS Cell of Military College of Signals has been used as a benchmark for threshold. The data taken from MIS Cell comprises over a periods of 12 months from June 2013 to June 2014. On average, there are 252 unique users according to the data. Figure 5.6 (a) and (b) illustrates the data of MIS Cell. It can be seen that the peak number of packet per second 1445 and the 2nd highest peak is at 482.

For test-bed 1, the average of all the values including the highest peak has been taken and used in Snort, Correlation and IAFV Algorithms. For Test-bed 2,the average of all the values excluding the highest peak value has been calculated and used in the algorithms. The reason of ignoring the highest peak in Test-Bed 2 is that in physical systems (used in Test-Bed 1), the maximum number of packets per second that Snort is able to receive is 5000, while in deter-lab (used in Test-Bed 2), the maximum numbers of packets per second that Snort is able to receive is 1200. After these number of packets, Snort begins to drop the packets.

5.10.1 Threshold for Snort

Rate filter parameter has already been discussed in Chapter 2. The "count" in the parameter is to be changed in order to change the threshold. Values of threshold are gauged in a way to find out detection capability in two situations, one with a moderate threshold and the other with a higher value of threshold.

In this way, the effect of changing threshold values on detection capability of Snort has been observed. Table gives the two thresholds that have been chosen:

Table 5.7 Thresholds for Snort

For Test-Bed 1		**For Test-Bed 2**	
Threshold 1	Threshold 2	Threshold 1	Threshold 2
76	1	29	1

5.10.2 Threshold for IAFV Algorithm

The following threshold value has been calculated for the two test-beds using the data of MIS Cell. The value is calculated using the IAFV formula explained in Chapter 3.

Attack Threshold for IAFV = 252-10/10 = 24.2

5.10.3 Threshold for Correlation and MSW-Correlation Algorithm

The correlation coefficient values of the packets per second has been calculated using MIS Cell network data. It has been found that correlation coefficient values do not fall below 0.003. Hence the threshold for attack is chosen to be 0.0029.

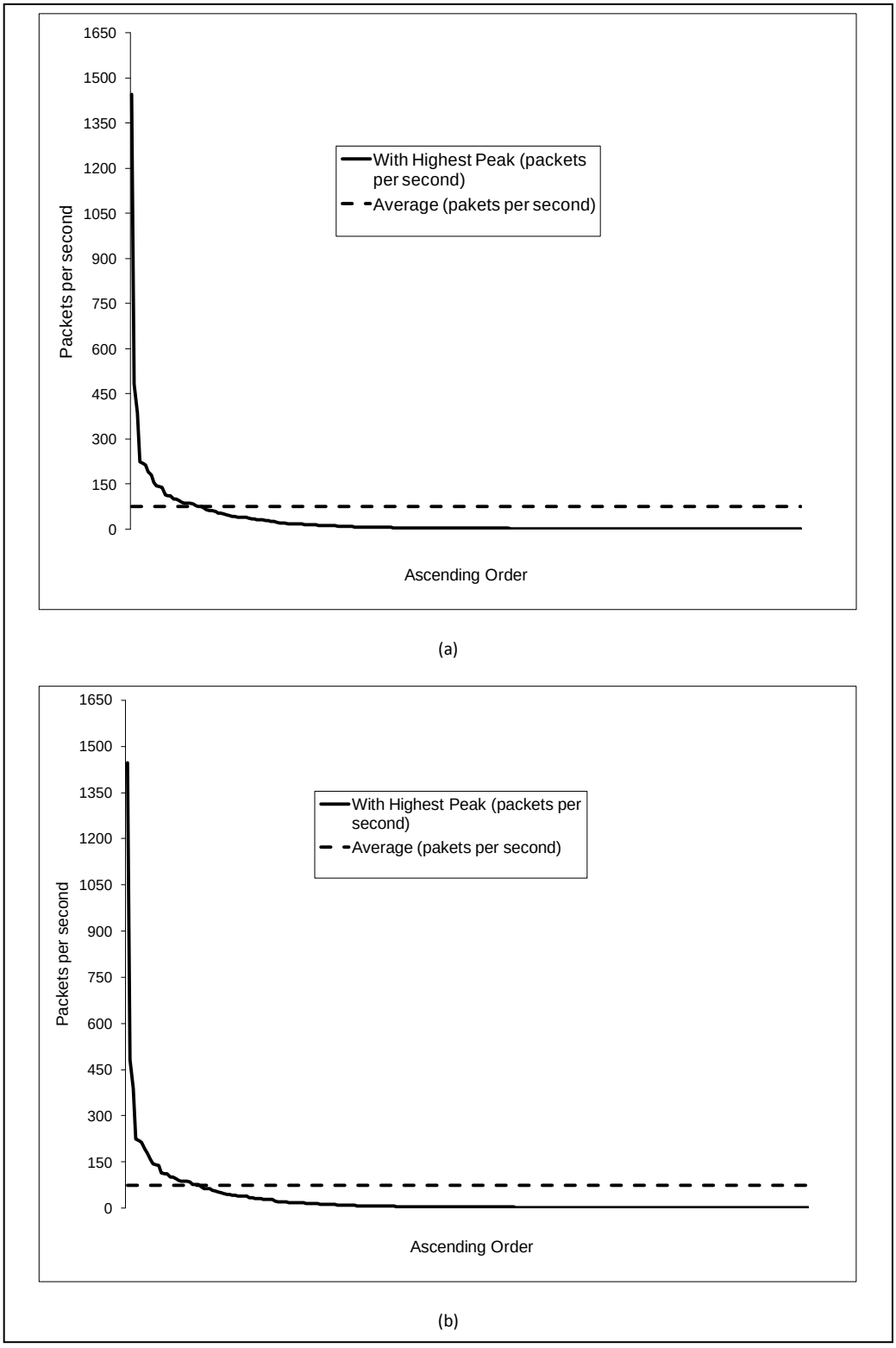

Figure 5.6 (a) Packet Per Second With Highest Peak (Ascending Order) (b) Packet Per Second Without Highest Peak (Ascending Order)

5.11 Conclusion

The datasets which are publicly available have been created more than a decade ago and thus are too old to be relied upon. The latest network trends are towards anonymous flooding DDoS attacks which are not reflected by these datasets.

This poses a limitation on usability of such datasets. A simpler and convenient alternative to these datasets is using traffic generator tools

available. Two test-beds have been used. In test-bed 1, network topology and network devices have been arranged physically according to the requirements of different test scenarios. In test-bed 2, network topology and network devices have been emulated using DETERlab.

While keeping the packet per second range steady, variations in the uniqueness of source IP addresses has been tested against both algorithms and for both test-beds. Tests have been done on attack traffic as well as on normal traffic. Based on the scenarios, we conducted the analyses with reference to detection capability, false negative and false positive ratio of the old and MSW-Correlation technique.

Chapter 6

Results And Analyses

6.1 Introduction

This chapter explains the results of the experiments conducted. According to the results, the proposed MSW-Correlation algorithm successfully identified the attack instances in all the attacks scenarios. The results have been shown below using graphs.

6.2 Results of Test-Bed 1(Design Using Real Systems)

This section explains the results that have been achieved in Test-bed 1 (please refer to Chapter 5 for test-beds details). Results for normal traffic scenarios have been explained in section 6.4.1 and the results of attack scenarios have been given in section 6.4.2.

6.2.1 Results of Normal Traffic Test Scenarios

This section explains the results of normal traffic scenarios belonging to test-bed 1. Please refer to Chapter No. 5 to read details about test-beds and traffic scenarios. The results of Snort, IAFV Algorithm and old Correlation versus MSW-Correlation technique have been given in sections 6.2.1.1, 6.2.1.2 and 6.2.1.3 respectively.

6.2.1.1 Results of Snort

Figure 6.1 shows the results of Snort when it receives normal traffic with different degree of unique source IP addresses according to the test scenarios. The x-axis shows

the normal test scenario numbers (for more details about scenarios, please refer to Table 5.2). Along the y-axis, 1 indicates occurrence of attack and 0 indicates vice-versa.

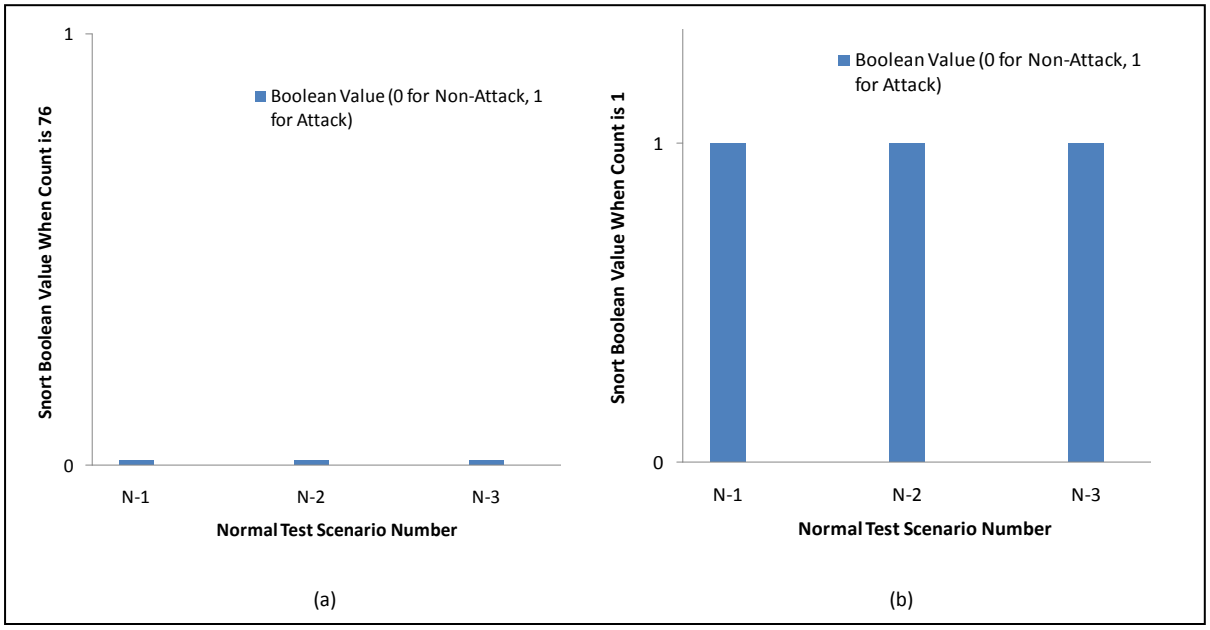

Figure 6.1: Snort Results For Normal Test Scenarios For Test-Bed 1 (a) When Count is 76 (b) When Count is 1 [Refer to Table 5.2 for Normal Test Scenarios]

6.2.1.2 Results of IAFV Algorithm

Figure 6.2 shows the detection capability of the IAFV technique. The threshold has been indicated using a straight line along the x-axis. Traffic flow with IAFV value greater than threshold is indicated as attack traffic (for more details about scenarios, please refer to Table 5.4).

6.2.1.3 Results of Old Correlation Technique Versus MSW-Correlation Technique

The results of different normal traffic scenarios have been shown in Figure 6.3. The mean of T1, T2, T3 and T4 are taken and denoted as "MSW-Correlation Technique". The

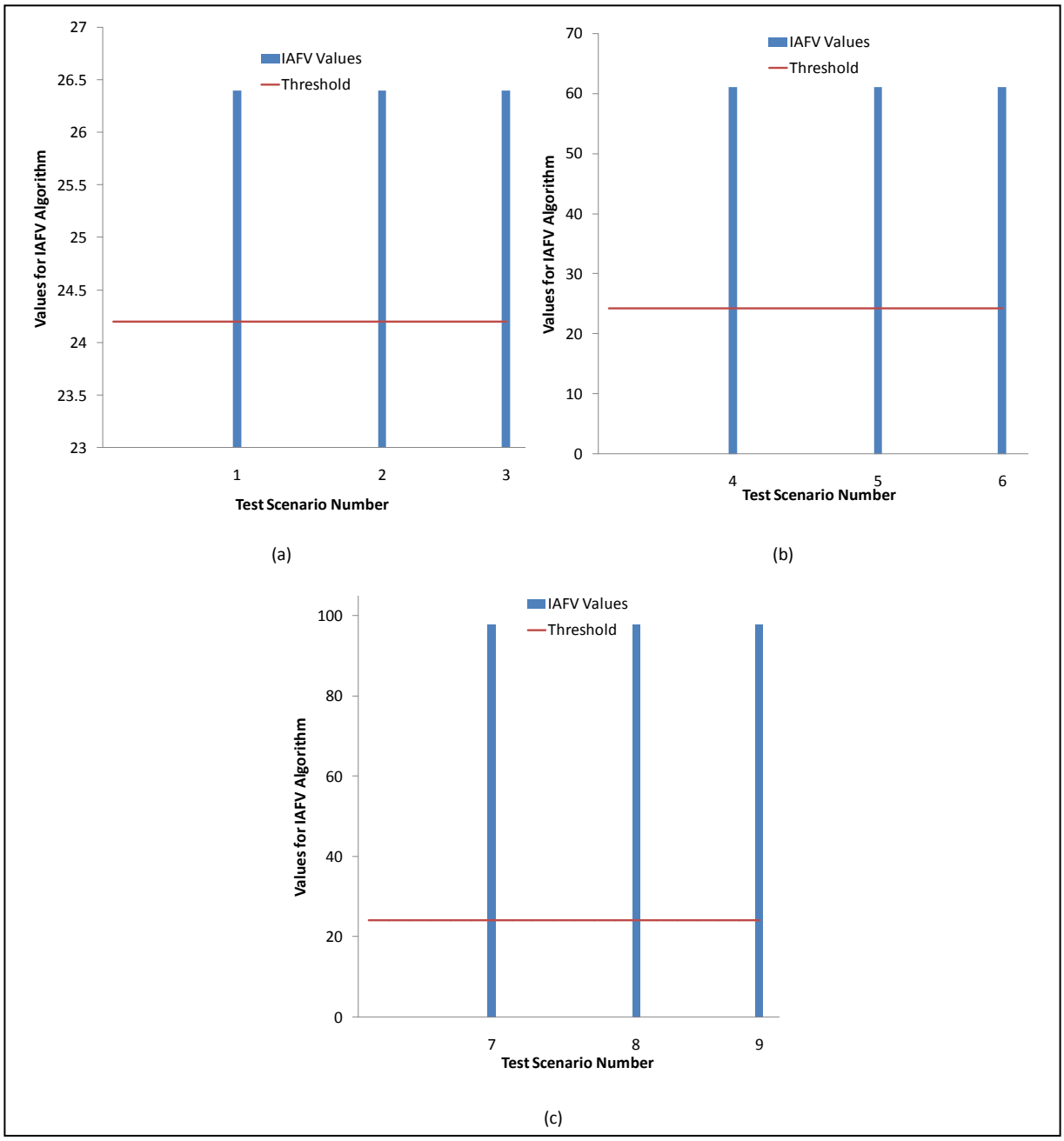

Figure 6.2: IAFV Results For Test-Bed 1 (a) For Test Scenarios 1,2 and 3 (b) For Test Scenarios 4,5 and 6 (c) For Test Scenarios 7,8 and 9[Refer to Table 5.4 for Test Scenarios]

x-axis shows the test scenarios and y-axis shows the csorrelation coefficient values respectively. The threshold has been indicated using a straight line along the x-axis. Traffic flow with correlation coefficient value smaller than threshold is indicated as attack traffic.

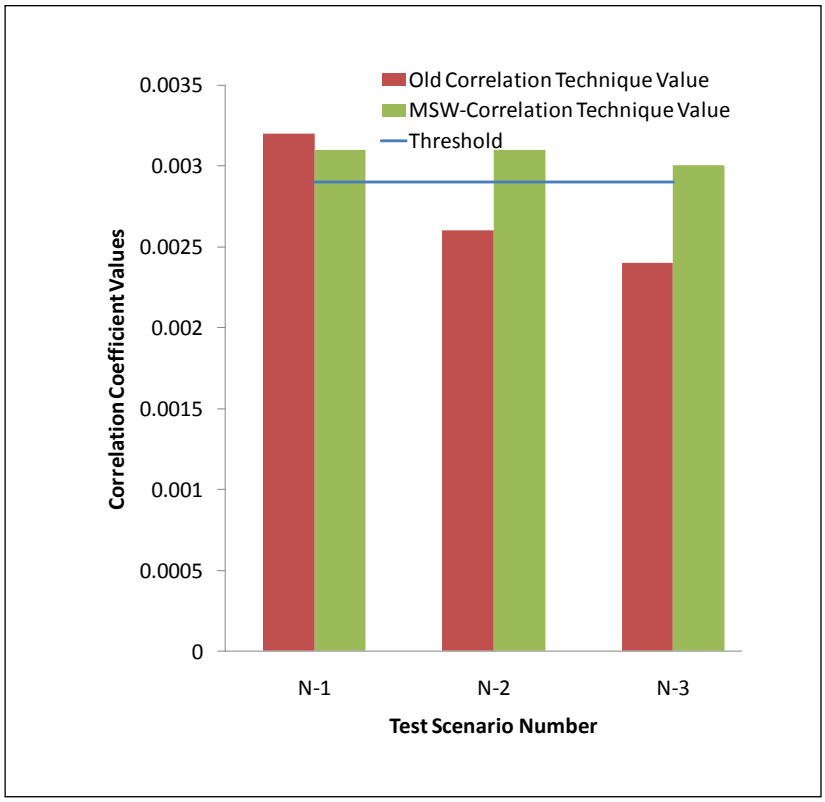

Figure 6.3: Old Correlation and MSW-Correlation Algorithms Results For Normal Test Scenarios for Test-Bed 1 [Refer to Table 5.2 for Normal Test Scenarios]

6.2.2 Results of Attack Traffic Test Scenarios

This section explains the results of attack scenarios belonging to test-bed 1. Please refer to Chapter No. 5 to read details about test-beds and traffic scenarios. The results of Snort, IAFV Algorithm and old Correlation versus MSW-Correlation technique have been given in sections 6.2.2.1, 6.2.2.2 and 6.2.2.3 respectively.

6.2.2.1 Results of Snort

Figure 6.4 shows the results for the detection capability of attack test scenarios when Snort IDS has been used as standalone flooding DDoS detection mechanism. The

decision of Snort whether attack has been launched or not is indicated along the y-axis by 0 or 1 respectively. Test scenarios have been mentioned along the x-axis.

6.2.2.2 Results of IAFV Algorithm

Figure 6.5 indicates the IAFV time series values. The x-axis shows the test scenarios and y-axis shows the IAFV values. The threshold has been indicated using a straight line along the x-axis. Traffic flow with IAFV value greater than threshold is indicated as attack traffic.

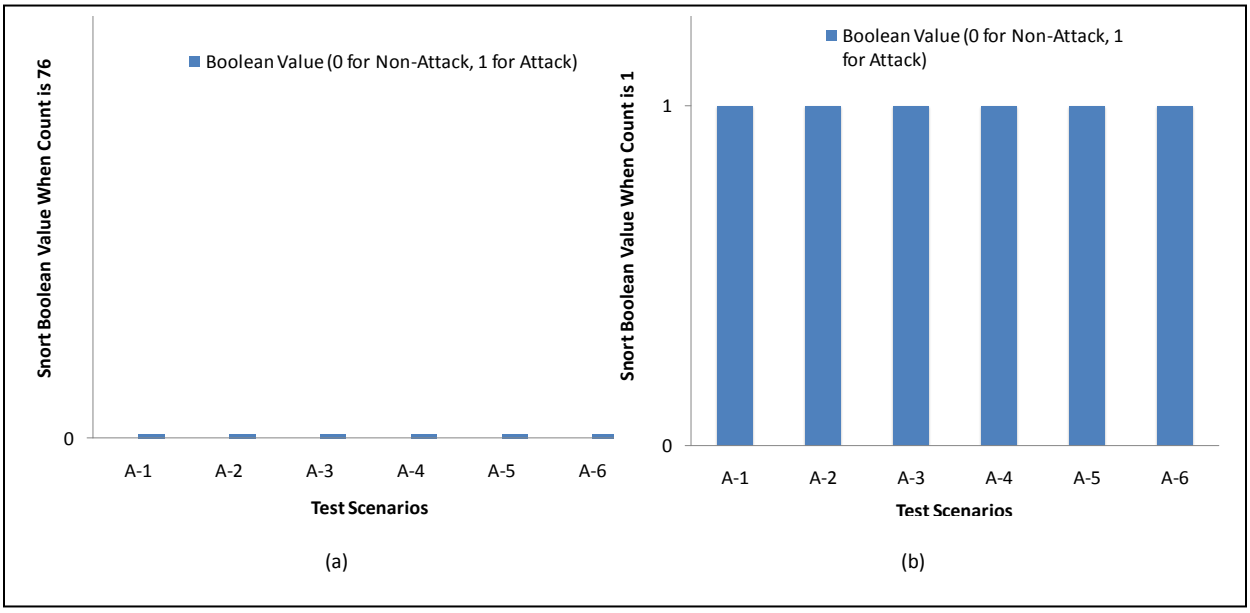

Figure 6.4: Snort Results For Attack Test Scenarios For Test-Bed 1(a) When Count is 76 (b) When Count is 1[Refer to Table 5.3 for Attack Test Scenarios]

6.2.2.3 Results of Old Correlation Versus Results of MSW-Correlation Algorithm

The results of different attack scenarios have been shown in Figure 6.6. The mean of T1, T2, T3 and T4 are taken and denoted as "MSW-Correlation Technique". The x-axis shows

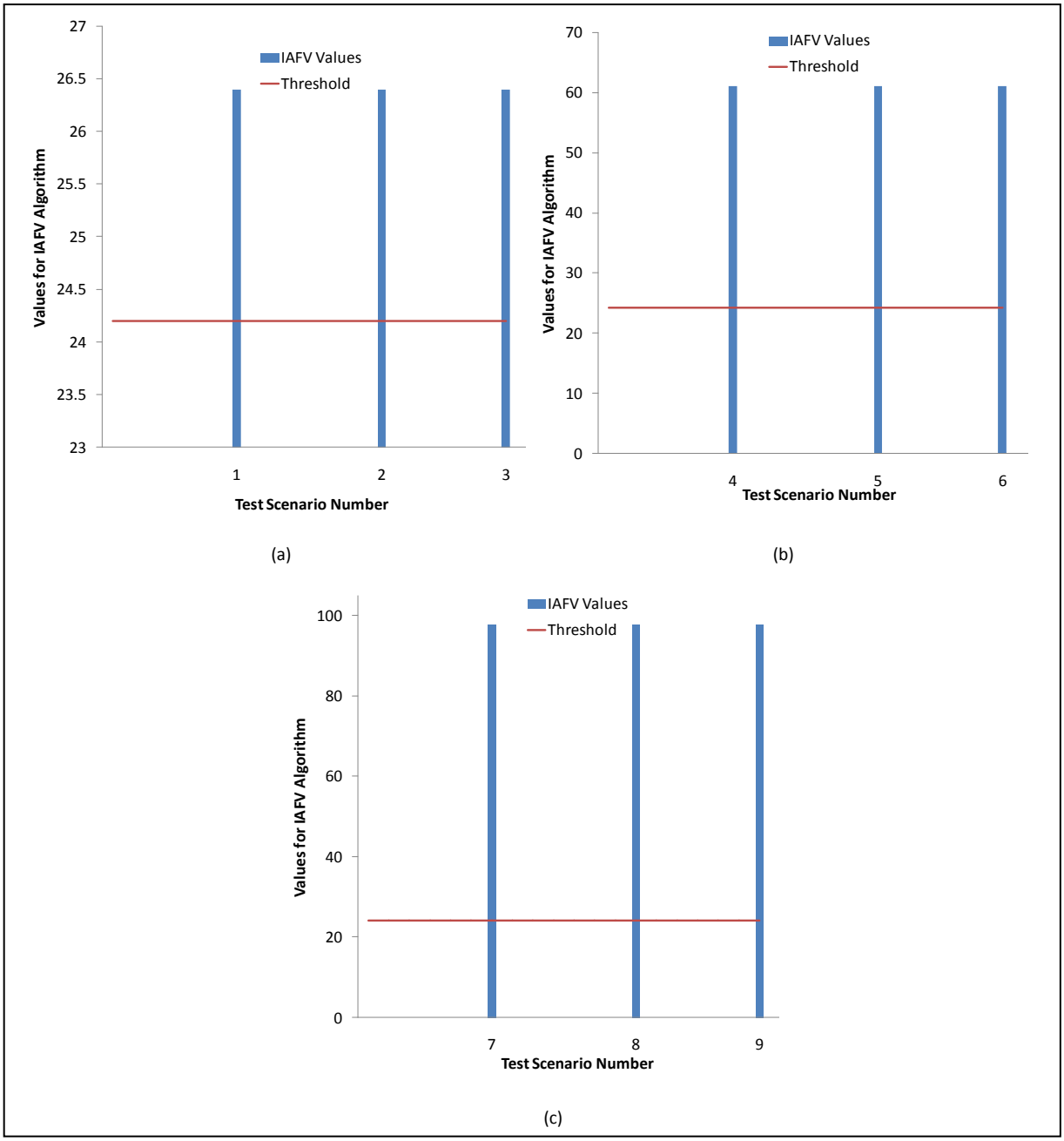

Figure 6.5: IAFV Results For Test-Bed 1 (a) For Test Scenarios 1,2 and 3 (b) For Test Scenarios 4,5 and 6 (c) For Test Scenarios 7,8 and 9[Refer to Table 5.4 for Test Scenarios]

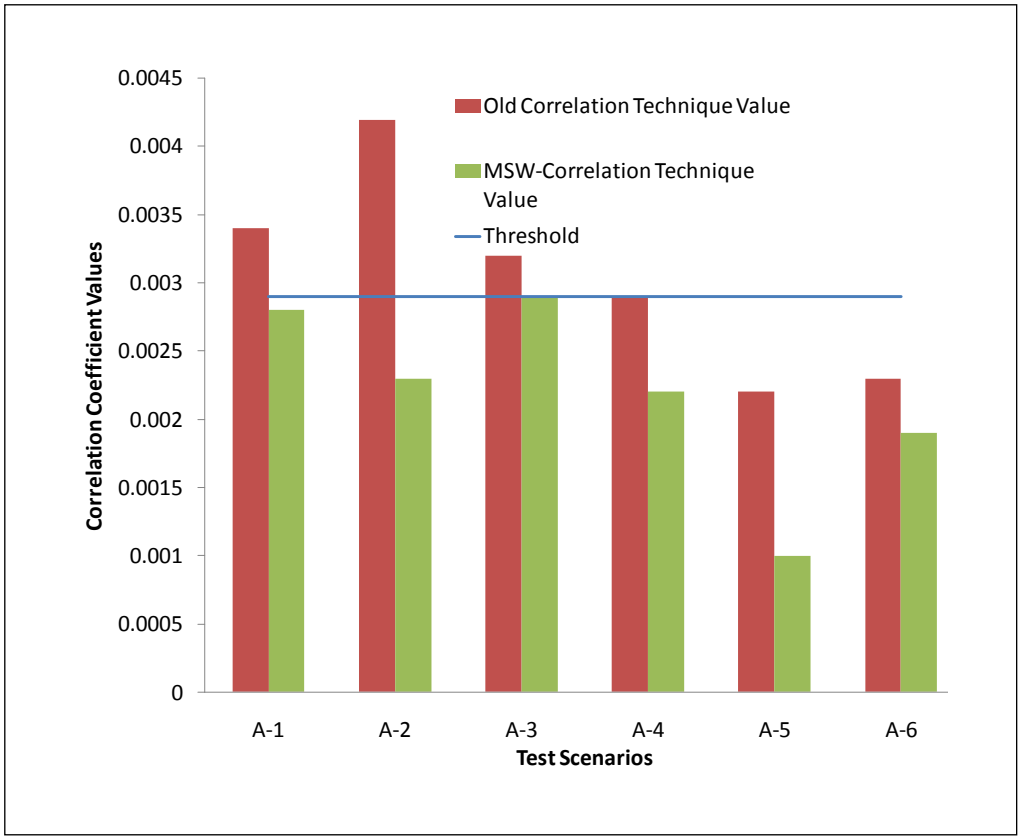

Figure 6.6: Old Correlation and MSW-Correlation Algorithm Results For Attack Test Scenarios for Test-Bed 1[Refer to Table 5.3 for Attack Test Scenarios]

the test scenarios and y-axis shows the correlation coefficient values respectively. The threshold has been indicated using a straight line along the x-axis. Traffic flow with correlation coefficient value smaller than threshold is indicated as attack traffic.

6.3 Results of Test-Bed 2 (Emulation Using DETERlab)

This section explains the results of both normal and attack traffic scenarios belonging to test-bed 2.

6.3.1 Results of Normal Traffic Test Scenarios

This section explains the results of normal traffic scenarios belonging to test-bed 2. Please refer to Chapter No. 5 to read details about test-beds and traffic scenarios. The results of Snort, IAFV Algorithm and old Correlation versus MSW-Correlation technique have been given in sections 6.3.1.1, 6.3.1.2 and 6.3.1.3 respectively.

6.3.1.1 Results of Snort

Figure 6.7 shows the results of Snort when it receives normal traffic with different degree of unique source IP addresses according to the test scenarios (refer to table 5.2).

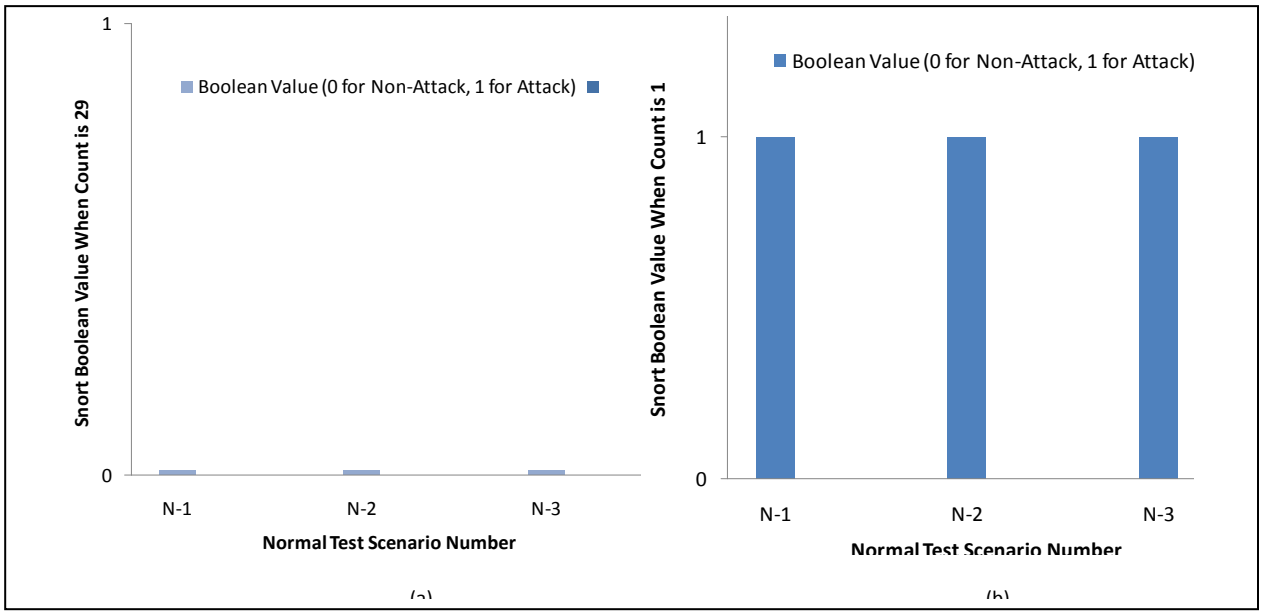

Figure 6.7: Snort Results For Normal Test Scenarios For Test-Bed 2 (a) When Count is 29 (b) When Count is 1

[Refer to Table 5.2 for Normal Test Scenarios]

6.3.1.2 Results of IAFV Algorithm

Figure 6.8 show the detection capability of the IAFV technique. The x-axis shows the test scenarios (refer to Table 5.4) and y-axis gives the IAFV values. The threshold has been indicated using a straight line along the x-axis. Traffic flow with IAFV value greater than threshold is indicated as attack traffic.

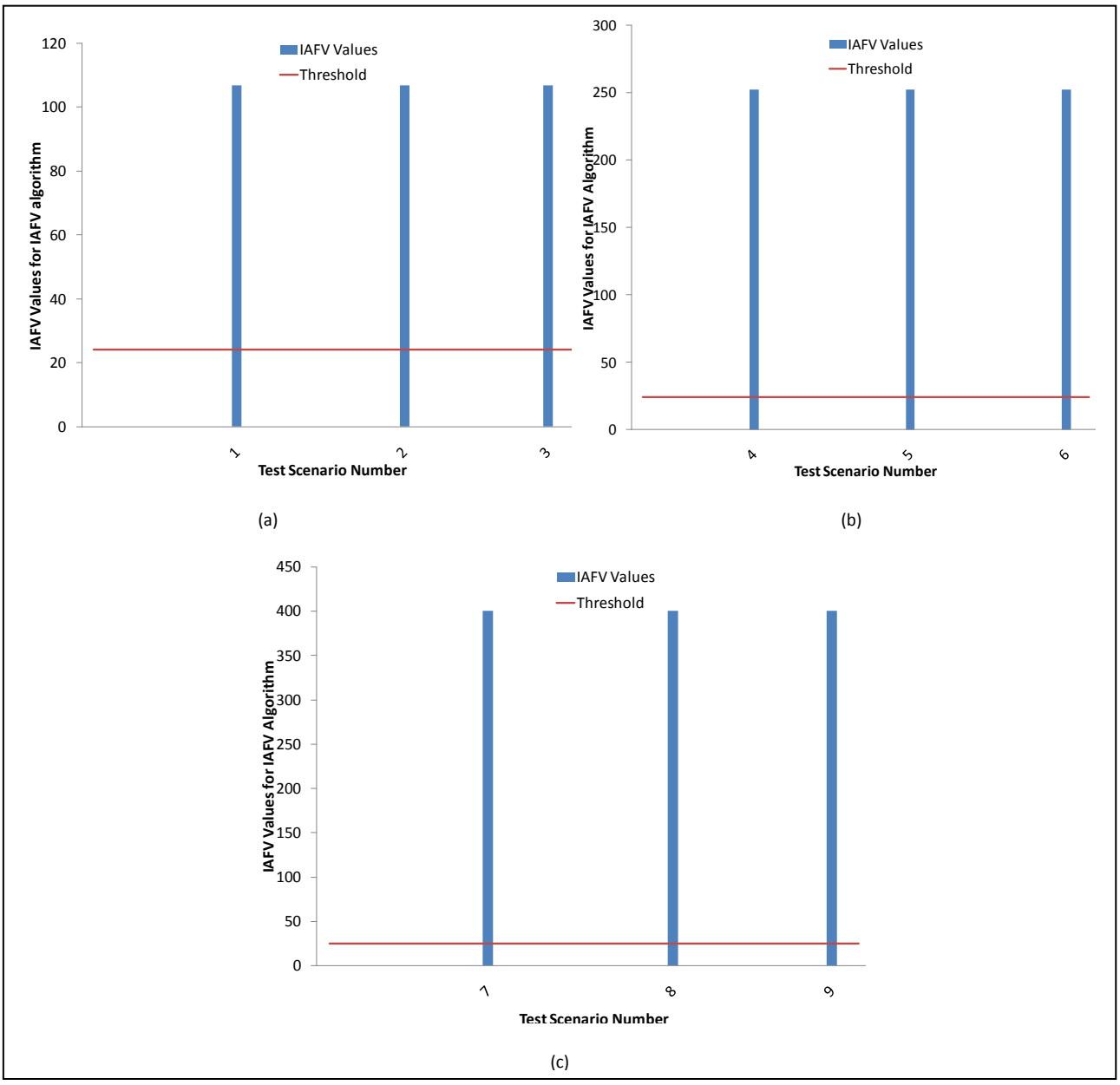

Figure 6.8: IAFV Results For Test-Bed 2 (a) For Test Scenarios 1,2 and 3 (b) For Test Scenarios 4,5 and 6 (c) For Test Scenarios 7,8 and 9[Refer to Table 5.4 for Test Scenarios]

6.3.1.3 Results of Correlation Technique Versus MSW-Correlation Technique

The results of different normal traffic scenarios have been shown in Figure 6.9. The mean of T1, T2, T3 and T4 are taken and denoted as "MSW-Correlation Technique". The x-axis shows the test scenarios (refer to Table 5.2) and y-axis shows the old and MSW-Correlation coefficient values respectively. The threshold has been indicated using a straight line along the x-axis. Traffic flow with correlation coefficient value smaller than threshold is indicated as attack traffic.

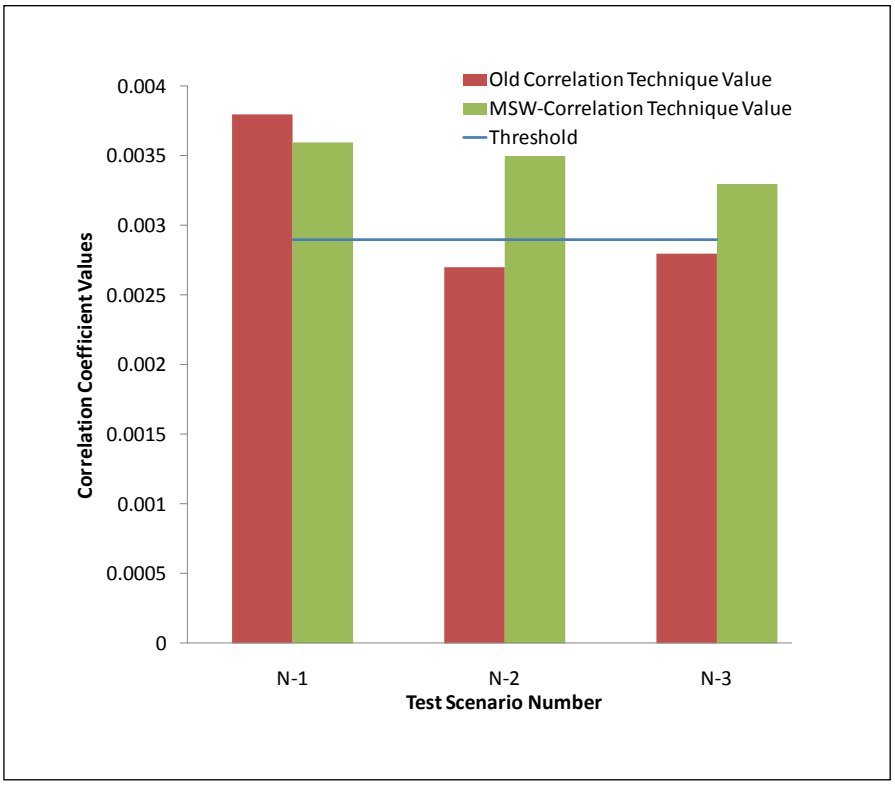

Figure 6.9: Old Correlation and MSW-Correlation Algorithm Results For Normal Test Scenarios for Test-Bed 2 [Refer to Table 5.2 for Normal Test Scenarios]

6.3.2 Results of Attack Traffic Test Scenarios

This section explains the results of attack scenarios belonging to test-bed 2. Please refer to Chapter No. 5 to read details about test-beds and traffic scenarios. The results of Snort, IAFV Algorithm and old Correlation versus MSW-Correlation technique have been given in sections 6.3.2.1, 6.3.2.2 and 6.3.2.3 respectively.

6.3.2.1 Results of Snort

Figures 6.10 shows the results for the detection capability of attack test scenarios when Snort IDS has been used as standalone flooding DDoS detection mechanism. The decision of Snort whether attack has been launched or not is indicated along the y-axis by 0 or 1 respectively.

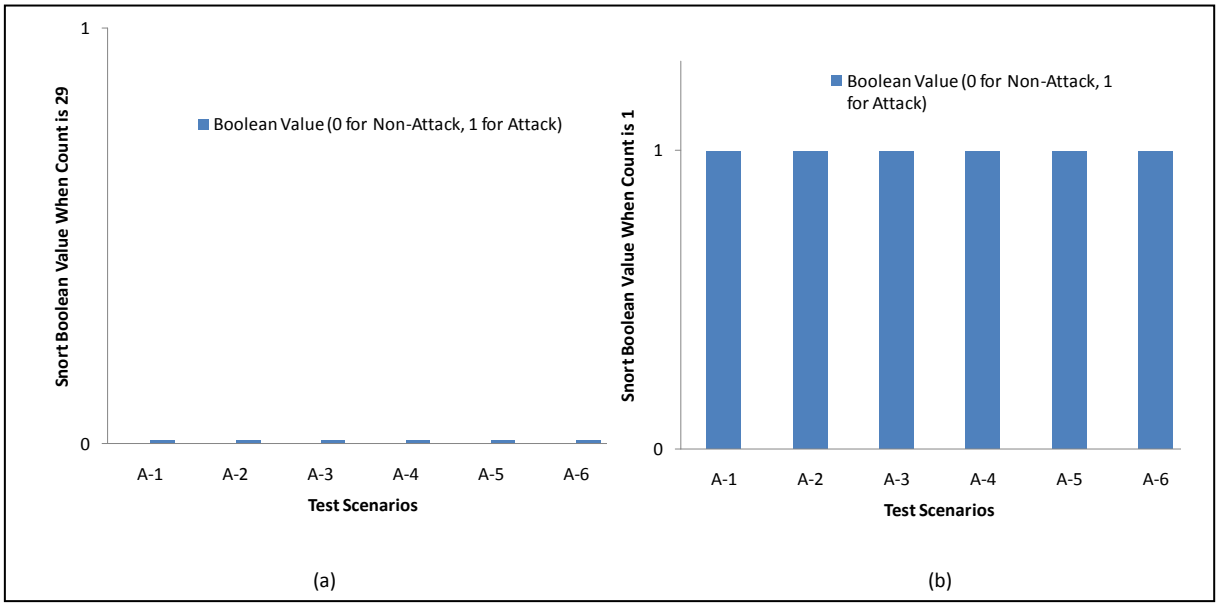

Figure 6.10: Snort Results For Attack Test Scenarios For Test-Bed 2(a) When Count is 29 (b) When Count is 1[Refer to Table 5.3 for Attack Test Scenarios]

6.3.2.2 Results of IAFV Algorithm

Figure 6.11 indicates IAFV time series values. The x-axis shows the test scenarios (refer to Table 5.4) and y-axis shows the IAFV values. The threshold has been indicated using a straight line along the x-axis. Traffic flow with IAFV value greater than threshold is indicated as attack traffic.

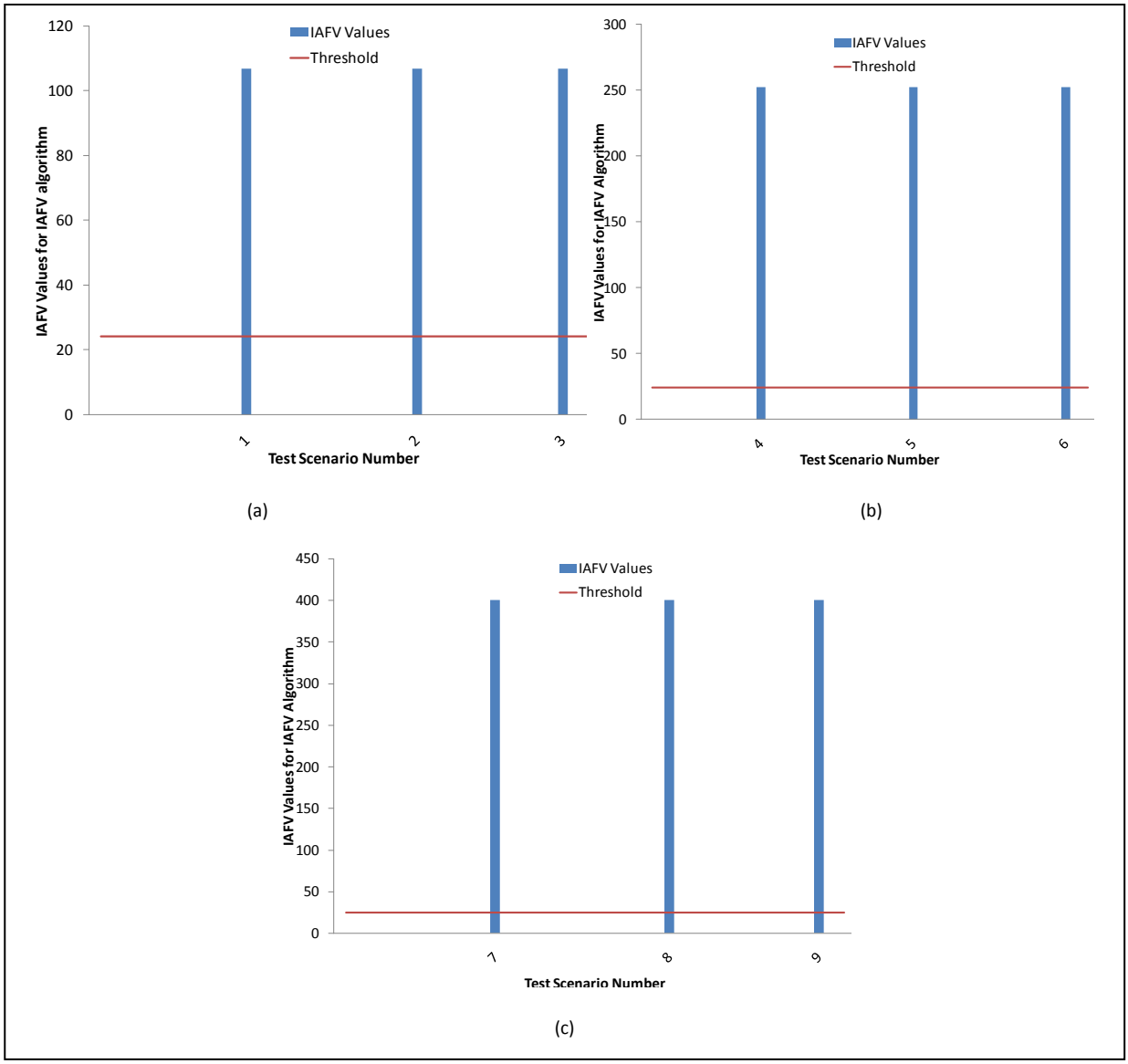

Figure 6.11: IAFV Results For Test-Bed 2 (a) For Test Scenarios 1,2 and 3 (b) For Test Scenarios 4,5 and 6 (c) For Test Scenarios 7,8 and 9[Refer to Table 5.4 for Test Scenarios]

6.3.2.3 Results of Old Correlation Versus Results of MSW-Correlation Algorithm

The results of different attack scenarios have been shown in Figure 6.12. The mean of T1, T2, T3 and T4 are taken and denoted as "MSW-Correlation Technique". The x-axis shows the test scenarios (refer to Table 5.3) and y-axis shows the old and MSW-Correlation coefficient values respectively. The threshold has been indicated using a straight line along the x-axis. Traffic flow with correlation coefficient value smaller than threshold is indicated as attack traffic.

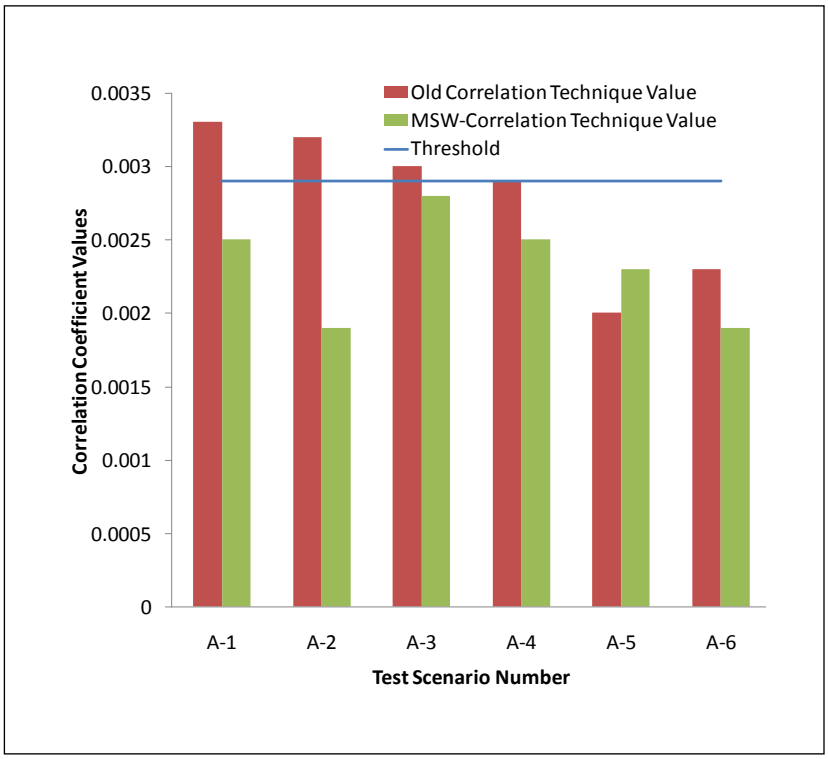

Figure 6.12: Old Correlation and MSW-Correlation Algorithm Results For Attack Test Scenarios for Test-Bed 2[Refer to Table 5.3 for Attack Test Scenarios]

6.4 Analyses

The following section gives a detailed analysis of the results of Test-bed 1 and Test-bed 2 with all the tested techniques under both attack and normal traffic test scenarios.

6.4.1 Results of Test-Bed 1 and Test-Bed 2

The following section gives a detailed analysis of the results of Test-bed 1 and Test-bed 2 with all the tested techniques under both attack normal traffic scenarios.

6.4.1.1 Analyses of Results of Snort

Figure 6.1 shows the detection capability of Snort under all the normal traffic scenarios in Test-bed 1 . As indicated in Figure 6.1 (a), Snort correctly detects the normal traffic as legitimate in all scenarios, when count is 76. On the other hand, in Figure 6.1 (b), it detects every incoming packet as attack in all scenarios, when count is 1.

Figures 6.4 shows the detection capability of Snort under all the attack scenarios Test-bed 1. With threshold 76 in Figure 6.4 (a), Snort is unable to detect attacks and it treats all packets as legitimate. While Figure 6.4 (b) indicates that when count is 1, Snort correctly detects attack packets in all scenarios. In short, if the threshold is chosen to be very small, every packet is detected as attack packet and vice versa if the threshold is chosen realistically.

Figure 6.7 shows the detection capability of Snort under all the normal traffic scenarios in Test-bed 2 . As indicated in Figure 6.7 (a), Snort correctly detects the normal traffic as

legitimate in all scenarios, when count is 29. On the other hand, in Figure 6.7 (b), it detects every incoming packet as attack in all scenarios, when count is 1.

Figures 6.10 shows the detection capability of Snort under all the attack scenarios Test-bed 2. With threshold 29 in Figure 6.10 (a), Snort is unable to detect attacks and it treats all packets as legitimate. While Figure 6.10 (b) indicates that when count is 1, Snort correctly detects attack packets in all scenarios. Hence, it can be concluded that Snort is unable to correctly differentiate between attack and legitimate traffic by its rate filter feature.

6.4.1.2 Analyses of Results of IAFV Algorithm

Figure 6.2 and 6.5 show the detection capability of IAFV algorithm under all the traffic scenarios in Test-bed 1. **Figure 6.8 and 6.11** show the detection capability of IAFV algorithm under all the traffic scenarios in Test-bed 2. Figures show that in all the traffic scenarios, IAFV is unable to differentiate between the destinations that are under attack and that are not under attack. It sums up the unique IP addresses and indicates that all the destinations as under attack. Hence, it can be concluded that IAFV time series method is unable to correctly differentiate between attack and legitimate traffic effectively.

6.4.1.3 Analyses of Results of Old Correlation Versus Results of MSW-Correlation Algorithm

Figure 6.3 shows the detection capability of old correlation algorithm and MSW-Correlation algorithm under all the normal traffic scenarios in Test-bed 1. **Figure 6.9**

shows the detection capability of old correlation algorithm and MSW-Correlation algorithm under all the normal traffic scenarios in Test-bed 2. The old correlation technique is unable to identify legitimate traffic in 2/3 of the scenarios. Although it is better than IAFV technique, still there is a room for improvement. According to the results, the MSW-Correlation technique is able to clearly identify the legitimate traffic. Hence, it outperforms the old correlation technique in both Test-beds 1 and 2.

Figure 6.6 and **Figure 6.12** show the detection capability of old correlation algorithm and MSW-Correlation algorithm under all the attack scenarios in Test-bed 1 and Test-bed 2 respectively. While it can be seen that the performance of old correlation technique is better than rest of the solutions discussed, but in both attack and test scenarios, the proposed technique outperforms the rest of the techniques and gives 100 detection rate.

6.4.2 Summary of Results For Test-Bed 1

The success of the proposed technique in Test-bed 1 is also depicted in **Table 6.1, 6.2 and 6.4**. It is shown that the MSW-Correlation technique correctly detects all the attack and legitimate packets. The crosses in the tables show the false negatives and ticks in the tables show true positive.

6.4.3 Summary of Results For Test-Bed 2

The results of Test-bed 2 are summarized in **Table 6.1, 6.3 and 6.5**. It is shown that the MSW-Correlation technique correctly detects all the attack and legitimate packets. The crosses in the tables show the false negatives and ticks in the tables show true positive.

Table 6.1: A summary of test scenarios in IAFV In Test-Beds 1 & 2 (refer to Table 5.4 for scenarios)

Destinations	Scenario Number								
	1	2	3	4	5	6	7	8	9
D1	×	×	✓	×	×	✓	×	×	✓
D2	×	×	✓	×	×	✓	×	×	✓
D3	×	×	✓	×	×	✓	×	×	✓
D4	×	×	✓	×	×	✓	×	×	✓
D5	×	×	✓	×	×	✓	×	×	✓
D6	✓	×	✓	✓	×	✓	✓	×	✓
D7	✓	×	✓	✓	×	✓	✓	×	✓
D8	✓	×	✓	✓	×	✓	✓	×	✓
D9	✓	×	✓	✓	×	✓	✓	×	✓
D10	✓	✓	×	✓	✓	×	✓	✓	×

Table 6.2: A summary of normal test scenarios in Snort, old correlation technique and MSW-Correlation technique for Test-Bed 1 (refer to Table 5.2 for scenarios)

Sc. No.	Test-Bed 1			
	Snort		Old Correlation Technique	MSW-Correlation Technique
	Threshold= 76	Threshold= 1		
N-1	✓	×	✓	✓
N-2	✓	×	×	✓
N-3	✓	×	×	✓

Table 6.3: A summary of normal test scenarios in Snort, old correlation technique and MSW-Correlation technique for Test-Bed 2 (refer to Table 5.2 for scenarios)

Sc. No.	Test-Bed 2			
	Snort		Old Correlation Technique	MSW-Correlation Technique
	Threshold= 29	Threshold= 1		
N-1	✓	×	✓	✓
N-2	✓	×	×	✓
N-3	✓	×	×	✓

Table 6.4: A summary of attack test scenarios in Snort, old correlation technique and MSW-Correlation technique for Test-Bed 1 (refer to Table 5.3 for scenarios)

Sc. No.	Test-Bed 1			
	Snort		Old Correlation Technique	MSW-Correlation Technique
	Threshold= 76	Threshold= 1		
A-1	×	✓	×	✓
A-2	×	✓	×	✓
A-3	×	✓	×	✓
A-4	×	✓	×	✓
A-5	×	✓	✓	✓
A-6	×	✓	✓	✓

Table 6.5: A summary of attack test scenarios in Snort, old correlation technique and MSW-Correlation technique for Test-Bed 2(refer to Table 5.3 for scenarios)

Sc No.	Test-Bed 2			
	Snort		Old Correlation Technique	MSW-Correlation Technique
	Threshold= 29	Threshold= 1		
A-1	×	✓	×	✓
A-2	×	✓	×	✓
A-3	×	✓	×	✓
A-4	×	✓	✓	✓
A-5	×	✓	✓	✓
A-6	×	✓	✓	✓

6.4.4 False Alarms

The analyses of all the algorithms has also been done by counting the false alarms each algorithm gives. As indicated in previous sections, the MSW-Correlation technique gives promising results and can distinguish between attack and normal traffic most effectively. Therefore, it gives least false alarms. This has been shown in Figure 6.13.

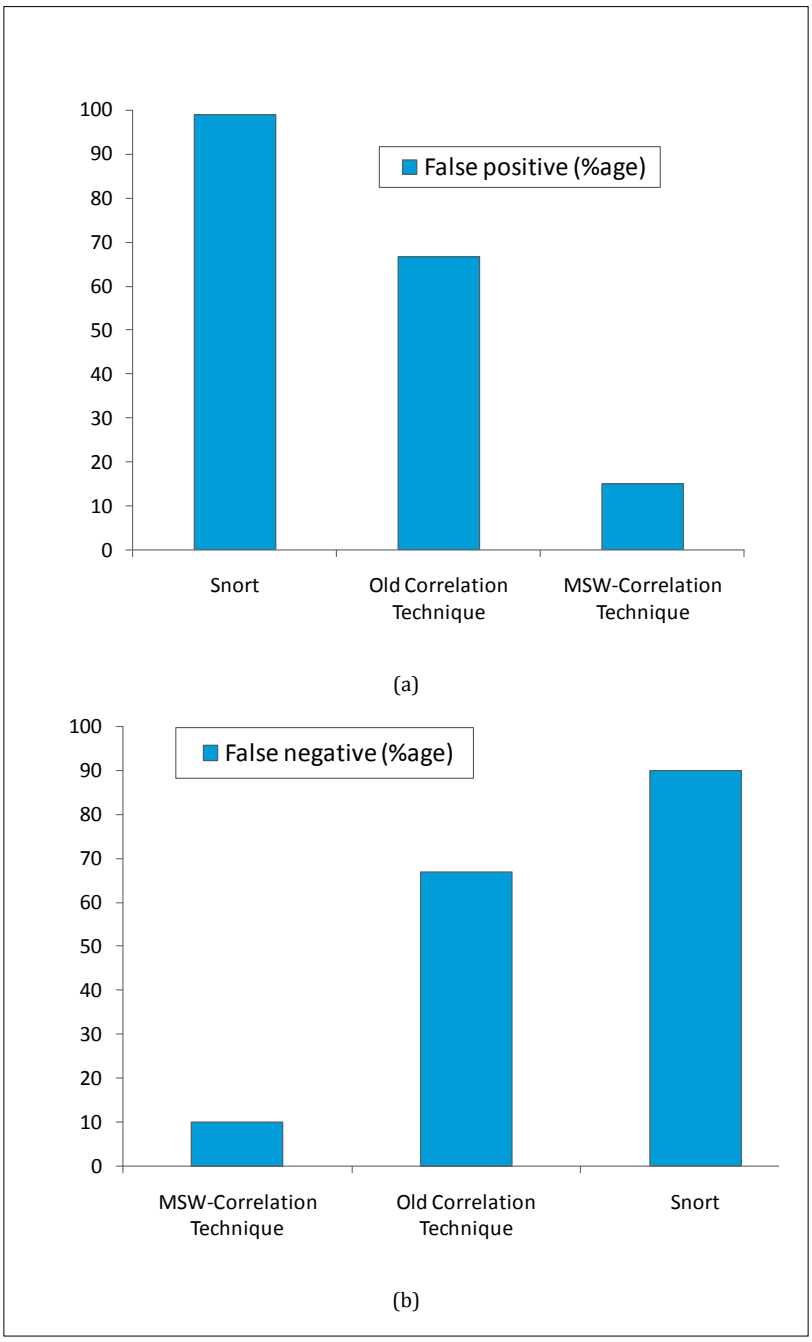

Figure 6.13: (a) False Positives (b) False Negatives

6.5 Conclusion

So far the conducted research in the field of detecting DDoS attack has been a challenging research problem since these attacks must be detected timely and

accurately. The main issues with most of the DDoS detection schemes has been that either they are not scalable or not accurate. The primary contribution of this chapter has been the results that are extracted from different attack and normal traffic scenarios. The results have been analyzed on the basis of detection accuracy and false alarms.

A comparison of the IAFV technique, old correlation technique, proposed technique and Snort has been given. Results indicate that the rate filter feature of Snort might be useful if the attack is launched from single source IP address ,which is generally not the case in flooding DDoS attacks. Practically, in the flooding distributed denial of service attacks, the attack traffic is generated using random source IP addresses. Hence, all the packets in our scenarios bypassed this feature which is the only feature in Snort to defend against flooding DDoS attacks. As indicated by results, the MSW-Correlation technique gives promising results and can distinguish between attack and normal traffic effectively.

Chapter 7

Conclusion

7.1 Overview

Flooding DDoS attacks are the most difficult attacks to detect timely and accurately. Unfortunately, to address this problem, the rate filtering technique in the present rule-based NIDS is insufficient because the packets sent seem to be legitimate. Also, the packet data does not match with any of the signatures in the NIDS database. Since the sources of flooding DDoS attacks are distributed or have been produced using tools that makes the attack look like coming from several thousand unique sources, it is very easy to bypass rate filters and limitations. The reason is that a very strict and low value of rate filter gives false positives and thus attacks will not be detected accurately. On the other hand, a higher value of rate filter will give false negatives and detect even the legitimate traffic as attack traffic as discussed previously.

7.2 Objectives Achieved

1. The detection techniques used by rule-based NIDS for flooding distributed denial of service attacks have been studied. Detection capability of chosen NIDS, Snort has been observed and analyzed in details with respect to the normal and attack scenarios in terms of false negatives and false positives. It has been seen from the experiments that, by keeping a low value of rate filter, the attacks have been detected in half of the attack scenarios , but at the same time, the legitimate

traffic was detected as attack traffic. While attack detection is the main motivation of rate filter, it should not detect normal traffic as attack. This will interrupt legitimate clients and cause denial of service. In this way, every incoming traffic, whether attack or legitimate, was detected as attack traffic in most of the scenarios. Thus, the detection capability of rate filter technique is found to be severely insufficient.

2. To generate effective results, a sophisticated test bench has been utilized. Both of the algorithms have been analyzed under several normal and flooding DDoS attack scenarios and evaluation has been done with respect to their detection accuracy and capability.
3. It has been observed in various recent studies that in order to detect flooding DDoS attacks, flow based techniques give much more promising results than packet based detection techniques. A variety of flow-based DDoS detection algorithms have been studied. Weaknesses in the present flow based DDoS detection techniques have been identified. The flow-based DDoS attack detection techniques have been divided broadly into two categories namely, Packet Based and Mathematical Formulation Based. Analyses has been done on two recent techniques one belonging to first category, IP Address Feature Value (IAFV) and the other belonging to second, Correlation of IP addresses.
4. It was found through experimental results that correlation technique is better than IAFV as it gives lesser false alarms. Hence, correlation technique was chosen

for further improvements that were expected be helpful in giving better results than both of the algorithms.

5. The proposed technique was implemented and integrated with a famous rule-based network intrusion detection system, Snort. The behavior or Snort was evaluated and then the effects of the integrated algorithm were evaluated to see the impact of the proposed technique. To the best of our knowledge, no flow based flooding DDoS detection technique has been integrated with Snort.

7.3 Limitations

During the course of the research, few limitations have been observed as follows:

1. The proposed correlation technique is currently using individual feature of packet header, i.e. source IP address.
2. The proposed technique has been tested on a limited number of real world datasets.

7.5 Future Directions

1. An obvious step forward would be to take several features and correlate them together. In case of multiple features, weights might be assigned to each feature and weighted correlation might be performed. This step is expected to increase detection capability potentially.
2. The proposed technique can be applied to a wider range of datasets comprising of more complex flooding attack types. This will help to generalize this technique.

7.6 Concluding Remarks

In this information technology based society, flooding DDoS attacks pose serious challenges to industries like media, entertainment, software, technology, security, financial services and gaming industries. Rule-based detection despite being the most common method suffers from limitations as it cannot monitor traffic flow and thus cannot detect flooding DDoS attacks efficiently. This thesis has made an attempt to handle this issue. Results have verified that the proposed technique is effective in detecting not only the attack traffic timely but also in reducing false positive rate. The technique has been integrated with rule-based NIDS, Snort. The proposed technique should be extended to deal with its explained limitations and future directions.

References

[1] "Internet Users in Pakistan", [Online] Available: http://tribune.com.pk/story/567649/30m-internet-users-in-pakistan-half-on-mobile-report/

[2] Monowar, Bhuyan, H. J. Kashyap, D. K. Bhattacharyya and J. K. Kalita, “Detecting Distributed Denial of Service Attacks: Methods, Tools and Future Directions”, December 2012

[3]"Cyber Security Statistics" [Online] Available: shttp://hackmageddon.com/category/security/cyber-attacks-statistics

[4] "Biggest DDoS Cloudfare", [Online] Available: http://rt.com/news/biggest-ddos-us-cloudflare-557/

[5] J. Mirkovic and P. Reiher "A taxonomy of DDoS attack and DDoS defense mechanisms", ACM SIGCOMM Computer Communication Review, vol. 34, no. 2, 2004

[6]"Power Systems Management and Associated Information Exchange—Data and Communications Security—Part 7", Network and System Management (NSM) Data Object Models; International Electrotechnical Commission (IEC): Geneva; Switzerland, 2009

[7]"Introduction to DDoS Attack by NSFOCUS", [Online] Available: en.nsfocus.com/ddos_faq/01-What_is_DDoS_Attack-EN.html

[8] "Dangerous DDoS (Distributed Denial of Service) on the rise" , [Online] Available: http://resources.infosecinstitute.com/dangerous-ddos-distributed-denial-of-service-on-the-rise/

[9] "Layer seven DDoS Attacks", [Online] Available: http://resources.infosecinstitute.com/layer-seven-ddos-attacks/

[10] "Q1 2014 Global Attack Report" [Online] Available : http://www.prolexic.com"

[11] "Akamai Publishes Prolexic Q1 2014 Global DDoS Attack Report" [Online] Available: http://www.akamai.com/html/ about/press/ releases /2014/press041714.html

[12] "DDoS Report 2014" [Online] Available : http://www.prolexic.com/knowledge-center-ddos-attack-report-2014-q1.html

[13] A. Saboor, M. Akhlaq, B.Aslam, "Experimental evaluation of Rule-based NIDS against DDoS attacks under different hardware configurations", In Proceedings of 2nd National Conference on Information Assurance (2013)

[14] H. Alaidaros, M. Mahmuddin and A. Al Mazari , “An Overview Of Flow-Based And Packet-Based Intrusion Detection Performance In High Speed Networks”, in Proceedings of the International Arab Conference on Information Technology (ACIT 2011), Riyadh, 2011

[15] A. Sperotto, “An Overview of IP Flow-Based Intrusion Detection,” IEEE Communications Surveys Tutorials, vol. 12, no. 3, 2010

[16] J. Vykopal, "Flow-based Brute-force Attack Detection in Large and High-speed Networks", Ph.D. dissertation, Masaryk University, 2013

[17] A. Sperotto, "Flow-based intrusion detection," Ph.D. dissertation, University of Twente, October 2010

[18] “Rule-based NIDS” [Online]. Available: www.Rule-based NIDS.org

[19] "Network Intrusion Detection and Mitigation against Denial of Service Attack" , [Online]. Available: www.cis.upenn.edu/~lindong/paper/wpe2.pdf(2013)

[20] P. Barford and D. Plonka, "Characteristics of Network Traffic Flow Anomalies" , In Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement, pages 69–73. ACM, 2001

[21] "DDoS protection flow detection overview", [Online]. Available: http://www.juniper.net/techpubs/en_US/junos13.3/topics/concept/subscriber-management-scfd-overview.html

[22] "The Bro Network Security Monitor ", [Online]. Available: www.bro.org/

[23] Gavrilis, D. and Dermatas, "Real-time detection of distributed denial-of-service attacks using RBF networks and statistical features", Computer Networks and ISDN Systems, 48, 235–245,2005

[24] J. Li, Y. Liu, and L. Gu, “DDoS Attack Detection Based On Neural Network,” Proc. of 2nd Intl’ Symposium On Aware Computing (ISAC), IEEE, pp. 196-199, November 2010

[25] R. Karimazad and A. Faraahi, "An anomaly-based method for DDoS attacks detection using RBF neural networks" In proceedings of the International Conference on Network and Electronics Engineering, Singapore, pp. 44–48. IACSIT Press, 2011

[26] Kumar , P. A. R. and Selvakumar, S. (2011) Distributed denial of service attack detection using an ensemble of neural classifier. Computer Communication, 34, 1328–1341

[27] P. Agarwal, B. Gupta, S. Jain, and M. Pattanshetti, “Estimating Strength of a DDoS Attack in Real Time Using ANN Based Scheme,” Communications in Computer and Information Science, Springer, 2011, vol. 157, part 6, pp. 301-310.

[28] B. Gupta, R. Joshi, M. Misra, A. Jain, S. Juyal, R. Prabhakar, and A. Singh, “Predicting Number of Zombies in a DDoS Attack Using ANN Based Scheme,” Communications in Computer and Information Science, Springer, 2011, vol. 147, part 1, pp. 117-122.

[29] H. Xu, B. Chen, F. Yang, and F. Liu, “Fast Algorithm of Evolutional Learning Neural Network,” Proc. of Int’l Conf. On Intelligent Systems Design and Engineering Application (ISDEA), IEEE, pp. 262-265, January 2012.

[30] M. Aamir und A. Zaidi, "DDoS Attack and Defense: Review of Some Traditional and Current Techniques“. In proceedings of CoRR abs, 1401.6317, 2014.

[31] F. Lipson, “Tracking and Tracing Cyber-Attacks: Technical Challenges and Global Policy Issues,” CERT Coordination Center, Special Report: CMU/SEI-2002-SR-009, November 2002

[32] K. Kumar, L. Sangal, and A. Bhandari, “Traceback Techniques Against DDoS Attacks: A Comprehensive Review,” In proceedings of Conference On Computer and Communication Technology (ICCCT), IEEE, pp. 491-498, September 2011

[33] K. Subhashini, and G. Subbalakshmi, “Tracing Sources of DDoS Attacks in IP Networks Using Machine Learning Automatic Defence System,” International Journal of

Electronics Communication and Computer Engineering, vol. 3, issue 1, pp. 164-169, January 2012

[34] H. Beitollahi, and G. Deconinck, “Analyzing well-known countermeasures against distributed denial of service attacks,” Computer Communications, Elsevier, vol. 35, issue 11, pp. 1312-1332, June 2012.

[35] A. Siris and F. Papagalou, “Application of Anomaly Detection Algorithms for Detecting SYN Flooding Attacks,” in Proceedings of IEEE Global Telecommunications Conference (GLOBECOM’04), vol. 4, Dallas, USA, pp. 2050–2054, 2004.

[36] H. Wang, D. Zhang, and G. Shin, “SYN-dog: Sniffing SYN Flooding Sources,” in Proceedings of the 22th International Conference on Distributed Computing Systems. Washington, DC, USA: IEEE Computer Society, pp. 421–429, 2002.

[37*]* N. Ye*,* S. Vilbert, and Q. Chen*,* "Computer intrusion detection through EWMA for auto correlated and uncorrelated data," In proceedings of *EEE* Trans. Rel., vol. 52, no. 1, pp. 75–82, March 2003.

[38] A. Lakhina, M. Crovella, and C. Diot, “Diagnosing Network-Wide Traffic Anomalies,” in Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communications(SIGCOMM’04), pp. 219–230, 2004

[39] L. Huang, X. Nguyen, M. Garofalakis, and M. Hellerstein, “Communication-Efficient Online Detection of Network-Wide Anomalies,” in IEEE Conference on Computer Communications (INFOCOM’ 07) , pp. 134–142, 2007

[40] N. L. D. Khoa, T. Babaie, S. Chawla, and Z. Zaidi, “Network Anomaly Detection Using a Commute Distance Based Approach,” in International Conference on Data Mining Workshops (ICDW’10), pp. 943–950, 2010

[41] H. Ringberg, A. Soule, J. Rexford, and C. Diot, “Sensitivity of PCA for Traffic Anomaly Detection,” in Proceedings of the ACM SIGMETRICS’ 07, pp. 109–120, 2007

[42] Y. Kim, J. Y. Jo, and K. K. Suh, "Baseline profile stability for network anomaly detection," International Journal of Network Security, vol. 6, No.1, pp. 60–66, Jan 2008

[43] Wu, Y. C., Tseng, H. R., Yang, W., and Jan, R. H., "DDoS detection and trace-back with decision tree and grey relational analysis", International Journal of Ad Hoc and Ubiquitous Computing, 7, 121–136, 2011

[44] T. M. Gil and M. Poletto, "MULTOPS: a data-structure for bandwidth attack detection," in Proceedings of 10th Usenix Security Symposium, pp. 23-38, 2001.

[45] H. Rahmani, N. Sahli and F. Kammoun, "Joint entropy analysis model for DDoS attack detection" In proceedings of the 5th International Conference on Information Assurance and Security, 2009.

[46] K. Lu, D. Wu, and J. Fan, “Robust and efficient detection of DDoS attacks for large-scale internet,” Computer Networks: The International Journal of Computer and Telecommunications Networking, vol. 51, no. 18, pp. 5036–5056, Dec 2007.

[47] M. Handley, “Internet architecture WG: DoS-resistant internet subgroup report,” Technical Report, 2005.

[48] A. Lakhina, M. Crovella, and C. Diot, “Diagnosing network-wide traffic anomalies,” in proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, Portland, Oregon, USA, pp. 219–230, 2005

[49] “Mining anomalies using traffic feature distributions,” in proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, Philadelphia, Pennsylvania, USA, pp. 217–228, 2005

[50] C. Zhang, Z. Cai, Chen, X. Luo and J. Yin, "Flow level detection and filtering of low-rate DDoS" Computer Networks, 56, 3417–3431, 2012

[51] L. Hellemons, L. Hendriks, R. Hofstede, A. Sperotto, R. Sadre, A. Pras, "SSHCure: A Flow-Based SSH Intrusion Detection System. In proceedings of AIMS 2012. LNCS, vol. 7279, pp. 86–97. Springer, Heidelberg, 2012

[52] D. Cabrera, "Proactive detection of distributed denial of service attacks using MIB traffic variables-a feasibility study," pp. 609-622, 2001

[53] 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 pages, April 2007.

[54] A. Sanmorino, S. Yazid, “DDoS Attack Detection Method and Mitigation Using Pattern of the Flow”, IEEE International Conference of Information and Communication Technology, 2013

[55] Z. Wang, X. Wang , "DDoS attack detection algorithm based on the correlation of IP address analysis" In Proceedings of the 2011 International Conference on Electrical and Control Engineering (ICECE). Yichang, 2011

[56] J. Jung et al. Fast portscan detection using sequential hypothesis testing. In Proc. of the IEEE Symposium on Security and Privacy, 2004.

[57] "Deterlab, based on Emulab", [Online] Available: http://www.deterlab.net/

[58] Y. Gao, Z. Li, Y. Chen, "A DoS Resilient Flow-level Intrusion Detection Approach for High-speed Networks", 2006

[59] J. Cheng, J. Liu, "DDoS Attack Detection Algorithm Using IP Address Features" In Proceedings of FAW 2009. LNCS. Springer, Heidelberg (2009)

[60] J. Vykopal, "Flow-based Brute-force Attack Detection in Large and High-speed Networks", Ph.D. dissertation, Masaryk University, 2013

[61] K. Scarfone and P. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS). NIST", Recommendations of the National Institute of Standards and Technology. Retrieved online December 26, 2009"

[62] B. Claise, "Cisco Systems NetFlow Services Export Version 9. RFC 3954 (Informational)", October 2004

[63] D. Kumar, Dr C. Guru Rao, Dr M. Singh, Dr Satyanarayana, "A Survey on Defense Mechanisms countering DDoS Attacks in the Network", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 7, July 2013

[64] H. Monowar, J. Bhuyan, D. Kashyap, K. Bhattacharyya and K. Kalita, “Detecting Distributed Denial of Service Attacks: Methods, Tools and Future Directions”, December 2012

[65] O. Salem, A. Makke, J. Tajer, and A. Mehaoua, "Flooding Attacks Detection in Traffic of Backbone Networks," in LCN, 2011, pp. 441-449.

[66] E. Ahmed, A. Clark, and G. Mohay, "A Novel Sliding Window Based Change Detection Algorithm for Asymmetric Traffic", In NPC 2008 IFIP International Conference on Network and Parallel Computing, 2008, pages 168–175. IEEE, 2008.

[67] E. Ahmed, A. Clark, and G. Mohay, "Effective Change Detection in Large Repositories of Unsolicited Traffic", In Proceedings of the Fourth International Conference on Internet Monitoring and Protection, May 2009.

[68] H. Takada and U. Hofmann. "Application and Analyses of Cumulative Sum to Detect Highly Distributed Denial of Service Attacks using Different Attack Traffic patterns, April 2004", [Online] Available : http://www.ist-intermon.org/dissemination/newsletter7.pdf

[69] H. Liu and S. Kim, “Real-Time Detection of Stealthy DDoS Attacks Using Time-Series Decomposition”, in Proceedings of IEEE International Conference on Communications (ICC), 2010

[70] J. Mirkoviac, G. Prier, and P. Reiher, "Attacking DDoS at the source. Proceedings of the 10th IEEE International Conference on Network Protocols", Paris, France, 12-15 November, pp. 1092–1648. IEEE CS, 2002

[71] L. Chen,, "A new detection method for distributed denial-of-service attack traffic based on statistical test", Journal of Universal Computer Science, 15, 488–504, 2009

[72] D. Kale, IIProf. V. Bhosale, " Scrutiny of DDoS Attacks Defense Mechanisms", International Journal of Advanced Research in Computer Science & Technology (IJARCST 2014), Vol. 2 Issue 1, 2014

[73] L. Feinstein, D. Schnackenberg, R. Balupari, and D. Kindred, "Statistical Approaches to DDoS Attack Detection and Response", In proceedings of DARPA Information Survivability Conference and Exposition, volume 1, pages 303–314. IEEE., 2003.

[74] J. Buchanan, J. Graves, R. Macfarlane “A Methodology to Evaluate Rate-Based Intrusion Prevention System against Distributed Denial-of-Service”, In Cyberforensics 2011.

[75] " Downloads for Snort IDS",[Online] Available: https://www.snort.org/downloads

[76] M. Akhlaq, F.Alserhani, I.U.Awan, J.Mellor, A.J.Cullen,P. Mirchandani, "Virtualization Efficacy for Network Intrusion Detection Systems in High-speed Networks" in Weerasinghe, D. (ed.) IS&DF, vol. 41, pp. 26–41. Springer, Heidelberg, 2010

[77] F.Alserhani, M.Akhlaq, I.Awan, A.Cullen, J.Mellor, P.Mirchandani,"Evaluating Intrusion Detection Systems in High Speed Networks" In proceedings of 5th International Conference of Information Assurance and Security (IAS 2009). IEEE Computer Society, Los Alamitos, 2009

[78] "The UCI KDD Archive University of California, Department of Information and Computer Science", [Online] Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html,

[79] J. McHugh, "Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory", ACM Transactions on Information and System Security (TISSEC), 3(4):262–294, 2000.

[80] M. Tavallaee, E. Bagheri, W. Lu and A. Ghorbani, "A Detailed Analysis of the KDD Cup 99 Data Set", In Proceedings of the 2009 IEEE Symposium Computational

Intelligence for Security and Defense Applications (CISDA 09)", IEEE Computer Society, 2009

[81] "Layer 7 DDoS Attacks, OWASP, 2010", [Online]. Available: https://www.owasp.org/images/4/43/Layer_7_DDOS.pdf

[82] "LOIC Project, SourceForge", [Online]. Available: sourceforge.net/projects/loic/

[83] "Ddos: Survey of traceback methods," A.John and T. Sivakumar, in International Journal of Recent Trends in Engineering, Vol. 1, No. 2, May 2009

[84] "The DETER Testbed: Overview",[Online], Available: *w*ww.isi.edu/deter/docs/testbed.overview.pdf

[85] "Using the Deter Test-bed by Ted Faber, 24 January 2011",[Online] Available: www.isi.edu/deter/docs/DETER_Tutorial-TF-Jan2011.pdf

[86] "tcpreplay". [Online]. Available: tcpreplay.synfin.net/wiki/Download

[87] "hping3". [Online]. Available*:* www.hping.org/hping3.html

[88] "Ostinato". [Online]. Available: code.google.com/p/ostinato/

[89] "Wireshark" [Online]. Available: https://www.wireshark.org/download.html

[90] "Cisco Catalyst 2960 Series Switches". [Online]. Available: http://www.cisco.com/en/US/products/ps6406/index.html

[91] Cearns, "Design of An Autonomous Anti-DDoS Network (A2D2),", Thesis, University of Western Ontario, London, Canada, 2002

[92] C. Akyazi and A. S. E. Uyar, “Distributed Intrusion Detection using Mobile Agents against DDoS Attacks,” in Proceedings of 23rd International Symposium on Computer and Information Sciences (ISCIS ’08), Istanbul, 2008, pp. 1-6

[93] Rik Busschers, “Effectiveness of Defense Methods Against DDoS Attacks by Anonymous”, University of Twente, 2010

[94] "Snort Manual: Rate Filtering", [Online] Available: manual.snort.org/node19.html

[95] Prahlad Fogla Giorgio Giacinto Wenke Lee Roberto Perdisci, Da-vide Ariu, Mcpad: A multiple classifier system for accurate payload-based anomaly detection, Computer Networks 53 (2009), 864-881

[96] K. Wang and S. Stolfo. Anomalous payload-based worm detection and signature generation. In Recent Advances in Intrusion Detection (RAID), 2005

[97]"Prolexic DDoS Attack Report 2013", [Online], Available: http://www.prolexic.com/news-events-pr-increasing-size-of-individual-ddos-attacks-20-gbps-is-the-new-norm-2012-q3.html

[98] "Cyber Attack Statistics 2013 by HackGeddon", [Online]. Available: http://hackmageddon.com/2013-cyber-attacks-statistics/