CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

A Machine Learning Based Classification Technique to Detect DDoS Attack in Cloud Computing Environment

by

Abdul Moqeet

A thesis submitted in partial fulfillment for the degree of Master of Science

in the

Faculty of Computing
Department of Computer Science

2021

This research is proudly deditaced to all my beloved family (my mum, my father, my wife, my daughter, my sisters) and all my friends. Thank you for your constant compassion, encouragement, sacrifices, guidance and support.

CERTIFICATE OF APPROVAL

A Machine Learning Based Classification Technique to Detect DDoS Attack in Cloud Computing Environment

by

Abdul Moqeet

(MCS183056)

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Muhammad Aleem	FAST NUCES, Islamabad
(b)	Internal Examiner	Dr. Amir Qayyum	CUST, Islamabad
(c)	Supervisor	Dr. Qamar Mahmood	CUST, Islamabad

Dr. Qamar Mahmood
Thesis Supervisor
February 2021

Dr. Nayyer Masood
Head
Dept. of Computer Science
February 2021

Dr. Muhammad Abdul Qadir
Dean
Faculty of Computing
February 2021

Author's Declaration

I, **Abdul Moqeet** hereby state that my MS thesis titled "**A Machine Learning Based Classification Technique to Detect DDoS Attack in Cloud Computing Environment**" is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/abroad.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my MS Degree.

(Abdul Moqeet)
Registration No: MCS183056

Plagiarism Undertaking

I solemnly announce that the research work proposed in this research thesis titled "**A Machine Learning Based Classification Technique to Detect DDoS Attack in Cloud Computing Environment**" is solely my research work with no substantial contribution from any other individual. Tiny contribution / help anywhere it is taken has been properly recognised and the full study has been written by me.

I recognise the principle of zero tolerance against plagiarism of the HEC and the Capital University of Science and Technology. Therefore, as the author of the above-mentioned article, I claim that no part of my research has been plagiarised and that any content used as a reference is correctly referred to / quoted.

I undertake that if I am found guilty of any formal plagiarism in the above-mentioned thesis even after the award of MS Degree, the University retains the right to withdraw / revoke my MS degree and that HEC and the University have the right to publish my name on the HEC / University website on which the names of the students who have submitted plagiarised work are posted.

(Abdul Moqeet)
Registration No: MCS183056

Acknowledgement

I would like to thank the Most Merciful and Most Merciful of **Allah** the Almighty, who has bestowed upon me the talents, wisdom and eternity of my efforts to arrive here and to accomplish my study goal. I am very thankful to **Dr. Qamar Mahmood**, supervisor of my research thesis, who directed me to complete my research thesis.

(Abdul Moqeet)

Abstract

Cloud computing is a model for allowing easy, unlimited, on-demand network access to a public computing resource pool. The DDoS attack is one of the main threats to cloud users as it compromises cloud providers' services and makes them unavailable to legal customers. Machine learning techniques are capable of identifying DDoS attacks, but also provides prevention. Static and dynamic machine learning techniques are used to select most adaptive (correlated) features. Static attributes selection techniques are suitable in the dynamic nature of incoming traffic. Therefore, a dynamic attribute selection techniques is required. NSL-KDD dataset is used in our work. Our proposed machine learning DDoS classification technique is categorized into three modules. One is pre-processing in which features of the dataset are selected and normalized on a standard scale [0-1]. In the second module, We used correlation based feature selector (CFS) with BestFirst-Search to select most correlated features of the dataset. This technique reduces the number of features from 41 to 9. In the last module, different classifier are used to classify DDoS and normal traffic. Support Vector Machine (SVM), Random Forest (RF), K-Nearest Neighbor (KNN), Multilayer Perceptron (MLP), AdaBoost (AB), and Decision Tree (DT) classifiers are used in our research work. J48 and Random Forest (RF) has produced high attack detection rate up to 98.7% with very low false detection rate. Further, selected attributes are also classified on the basis of protocol; ICMP, TCP and UDP. J48 and Random Forest (RF) has produced high protocol based attack detection rate of 99.7% for TCP and UDP, and 96.5% for ICMP with very low false detection rate.

Contents

List of Figures

List of Tables

Abbreviations

AB	Adaboost Classifier
CfsSubsetEval	Correlance Based Bttribute Selection Technique
DDoS	Distributed Denial of Service Attack
EDOS	Economic Denial of Sustainability Attack
EMFFS	Ensemble-based Multi-feature Filter Selection Method
FDR	Fisher Discriminant Ratio
IAAS	Infrastructure as a Service
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
J48	Name of a Classifier
KNN	K-nearest Neighbor Classifier
MLP	Multilayer Perceptrons Classifier
MADM	Multivariate Similarity Analysis-Based Monitoring technique
NB	Naive Bayes Classifier
PAAS	Platform as a Service
PCA	Principal Component Analysis
RF	Random Forest Classifier
SAAS	Software as a Service
SOM	Self-organizing Map
SVM	Support Vector Machine Classifier
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
VM	Virtual Machine

Chapter 1

Introduction

Cloud computing is becoming popular in the business IT world because it offers cost-effectiveness and scalability [1]. You may use computational resources such as computer power, storage and data databases instead of buying, owning and maintaining physical data centers and servers. It is a concept of configurable computing resources that are available to everyone at any time, such as servers, networks, storage, applications and services. This pool of resources is exchanged in a virtualization way that makes it easy to reserve and free resources easily. There are three major service models [2], namely software-as-a-service (SaaS), platform as a service (PaaS) and infrastructure as a service (IaaS).

The DDoS attack is one of the most popular and significant cyber attack [1] in the recent history. The aim behind the launching the DDoS attack is to consume the resources of the victim. Attacker sends a huge amount of traffic to the victim's side. As a result, these services would not be used for a specific time and thus the service cannot be offered to legal customers. It is one of the most common and most frustrating challenge on both cloud providers and its users. Many well-known cloud vendors, such as Amazon EC2 and Rackspace, have suffered DDoS attacks in recent years [3], resulting in financial loss of thousands of dollars. These attacks are becoming more severe and dangerous day by day in term of long duration, huge volume of traffic. These attacks are very hard to detect and track because it

is difficult to differentiate between attack packets and legitimate packets as they are coming from distributed sources.

The CIA triad (Confidentiality, Integrity, and Availability) is a well-known model for implementation of security policies. DDoS attack may mainly compromise the availability of computing resources, resulting in financial loss or impacting credibility. There are many things that may jeopardize availability, including hardware or software failure, power failure, natural disasters, and human error. Probably the most well-known attack that affects availability is a denial-of-service attack in which the performance of a system, server, web-based application, or web-based service is deliberately and maliciously disrupted or the system becomes entirely unavailable. A Distributed Denial of Service (DDoS) attack is considered as being the greatest danger to the IT sector [4] and there is huge increase is observed in every year.

Malicious network threats have been on the rise over the last decade. One of the most destructive attacks, often carried out over DNS, is carried out by command and control, often called C2 or C&C. Botnets can be used to execute Distributed Denial-of-Service (DDoS) attacks, steal data, deliver spam, and allow the attacker to access and link to the computer. The user can manage the botnet using the C&C command and control program. Itact as command centers that use malware connected to targeted attacks to store compromised data or download commands. Establishing C&C connections is a critical measure for attackers to travel sideways within a network. C&C servers also act as headquarters for infected computers on the botnet. Command and Control environment is built to launch a DDoS attack. There are several infected computers involved in it. The first hacker discovers infected machines on the Internet, and these machines discover more malicious machines that are accessible on the Internet. There is a sequence of infected machines is built. If the attacker needs to initiate an attack on a targeted server, just issue an order to the infected computer to send a massive amount of traffic to the victim side. It results in consuming network resources on the victim side and make it inaccessible to the valid user. Command and control (C&C) environment in which target is Cloud is shown in figure 1.1.

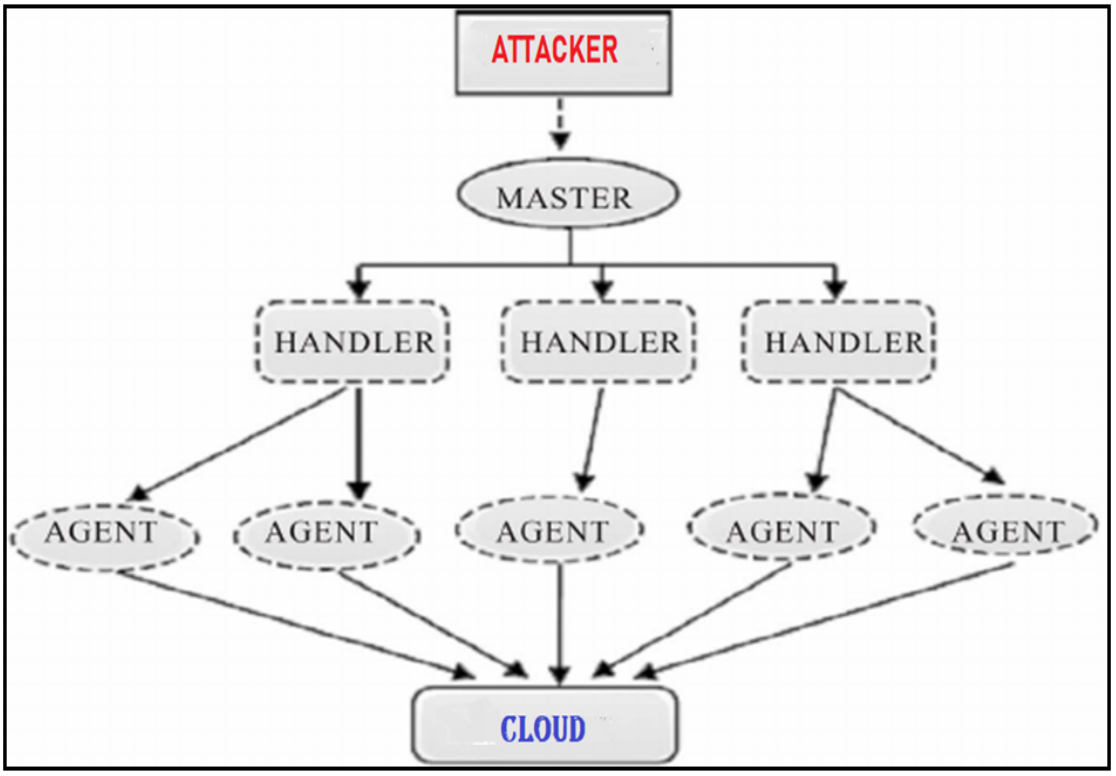

FIGURE 1.1: Command and Control Environment

Many variants of DDoS attacks exist in a cloud environment that differs in purpose, implementation strategy and scale. DDoS attacks can be narrowly categorized into two classes [5], brute-force and semantic. In brute-force attacks, attackers send a huge amount of malicious requests to exhaust the network bandwidth of the targeted cloud server. It is also known as flood/high-rate DDoS attacks. Attacker sends a stream of malicious requests either to interrupt cloud services or to disrupt users' connections. Connectivity disruption is caused by the overload of router processing capacity, network bandwidth capacity or resources. These attacks are referred to as Network or Transport layer flood attacks. The goal of an attackers is a sudden traffic jam that blocks the highway, stopping normal traffic from arriving at its destination. The attacks can easily detect by defense mechanisms due to the high rate of attack traffic. Examples of such attacks are given below.

- Transmission Control Protocol (TCP)-SYN flood
- User Datagram Protocol (UDP) flood
- Internet Control Message Protocol (ICMP) flood

- Hypertext Transfer Protocol (HTTP) flood
- Domain Name System (DNS) flood
- Simple Mail Transfer Protocol (SMTP) flood.

The attackers initiate such attacks by using the weakness to create attack armies, also known as botnet, of a maximum number of computers. An intruder sends the attack command to the C&C server, which is distributed to the several affected hosts [6]. The infected hosts send a storm of requests to one or more cloud servers.

Semantic attacks, on the other hand, exploit protocol vulnerabilities rather than consume network bandwidth or cloud storage capabilities. It's also classified as Vulnerability attacks. The attacker produces a low amount of malicious traffic directed at a given protocol or program. Such attacks are known as low-level DDoS attacks. Low-rate attack traffic is close to legal traffic. A low-speed DDoS attacker exploits the weakness of TCP's congestion-control system by regularly sending burst attack packets over a short period of time repeatedly (pulsing attack) or constantly launching attack packets at a constant low-rate rate (constant attack). Unlike the high-rate DDoS attack, the low-rate DDoS attack is a sophisticated and difficult to track due to its low-speed traffic and stealthy behavior. It is also difficult to distinguish a low-rate DDoS attack relative to a high-rate DDoS attack. Because the attacker sends malicious requests at a very low pace [7], the protection mechanisms based on traffic volume remain undetected. These attacks are listed below.

- **Shrew attack:** The intruder uses man-in-the-middle strategy for a low rate DDoS attack on the Transmission Control Protocol.
- **RoQ attack:** Reduction of Quality is a low-rate attack goal to devalue the quality of the target network(s).
- **LoRDAS:** A low-rate DoS attack against application servers.
- **EDoS:** Economic Denial of Sustainability attack exploits cloud elasticity and auto-scaling capabilities.

In High-rate DDoS attack, TCP, UDP and ICMP flooding attacks are the most common and severe attacks [5] to the cloud computing environment. Our area of interest is TCP, UDP and ICMP flooding attacks. The spoofed IP addresses are used by the attacker. The Internet Control Message Protocol (ICMP) defined by RFC 792 [8] is used to monitor network errors. This attack is launched by sending a huge amount of ICMP traffic to the cloud server and never get responded to it because of spoofed IP addresses. The TCP (Transmission Control Protocol) as defined in RFC 793 [8] is a connection-oriented protocol that operates on both the Open Systems Interconnection (OSI) and the TCP/IP protocol layers. It uses a three-way handshake before it transmits data between the sender and the receiver. It guarantees that the data is transmitted across the network. But due to spoofed IP addresses, the attacker never gets ACK packets from the server which makes a halfway connection opens. The User Datagram Protocol (UDP) defined by RFC 768 [8] is a connectionless protocol, relating to the Transportation Layer of both the Open Systems Interconnection (OSI) model and the TC / IP protocol stack. However, unlike the Transmission Control Protocol (TCP), the User Datagram Protocol (UDP) does not offer any guarantee for the transfer of datagram packets to the recipient. The attacker sends a stream of UDP traffic on random ports of targeted machine and these random ports may or may not unable. Cloud server is unable to respond these request because of spoofed IP addresses.

There are 34 surveyed techniques related to DDoS attack detection in a cloud computing environment. These techniques include signature based detection system [9], anomaly based detection [10], machine learning based techniques [11], [12] and software defined networking (SDN) [13]. Signature based IDS provides fast detection, but unable to detect unknown attacks. Anomaly based IDS are a bit slower than Signature based IDS and good to detect unknown attacks [10].

Machine learning based techniques [2], [14] and software defined networking (SDN) [15], [16], [17] provide DDoS detection as well as prevention. In machine learning based classification of DDoS attack, it is important to pick the most suitable and correlated attributes. Due to the growing volume of data that needs to be analyzed, feature selection may be used to classify essential features in a dataset,

with the goal of enhancing accuracy, precision, recall and f-score [2] and reducing computational complexity. Static attributes selection techniques are not capable of achieving accurate detection of DDoS attacks in dynamic environments [18]. Therefore, an adaptive attributes selection technique is required. We used correlation feature selection (CFS) techniques to select most correlated and the minimum number of attributes. In the SDN, a software-based traffic monitoring, centralized network management significantly enhances DDoS threat prevention [15] and mitigation capability.

There are different machine learning based classification techniques are discussed in literature [19], [20], [18]. Random Forest (RF), J48, Decision Tree (DT), K-nearest Neighbor, Naive Bayes (NB), AdaBoost (AB), Multilayer Perceptrons (MLP) are discussed. Classifiers classify the traffic in normal and anomaly class. These classifiers are also able to classify TCP, UDP and ICMP protocols [18].

1.1 Motivation

Several research reports on DDoS attacks and corresponding security methods in a cloud environment have been published. There are 48 techniques discussed in the literature which perform DDoS attack detection and prevention. Machine learning based techniques are capable of identifying high-rate DDoS attacks with maximum accuracy as compared to other discussed DDoS defensive techniques. Static attributes selection techniques are not suitable for dynamic nature of traffic. These techniques are needed to be improved to get maximum accuracy and low false detection of DDoS attack. Following are points to be considered in our research.

1. Identify the most relevant attributes and minimum number attributes to reduce time and space complexity for classification algorithms.

2. Classification of DDoS attack with different effective classifiers to improved accuracy, precision, recall and F-score.

1.2 Problem Statement

Machine learning based techniques which are discussed in literature are classified by selecting the maximum number of attributes present in the dataset. Less important attributes are needed to remove [18] as they increase time and space complexity. Therefore, a hybrid machine learning based classification technique is required, which selects minimum number of attributes to use limited system's resources and to classify the attack traffic and the normal traffic.

1.3 Research Questions

The Problem statement is raising research questions which are given below.

1. What are the limitations in surveyed machine learning techniques?
2. What is the minimum number of attributes which are used in our proposed machine learning based technique?
3. What are the parameters used by surveyed machine learning techniques to evaluate the performance of their technique?

1.4 Proposed Research Methodology

Research Methodology of our research is discussed as follows:

1. Explore research topic using different web resources.
2. Perform literature review to find the strength and limitations of surveyed techniques.
3. Analysis of normal and DDoS traffic to know the system' behavior.
4. Proposed machine learning based technique for DDoS detection.

5. Setting up an experimental environment for proposed machine learning technique.

6. Comparison of proposed technique results with already published technique [18].

1.5 Organization of Thesis

Chapter 2 discuss about literature review of the different defense techniques used to defend and protect DDoS attacks in the cloud environment. This chapter is divided into 4 sections; DDoS detection techniques and their limitations, DDoS prevention techniques and their limitation, comparative analysis of the techniques on the basis of entropy, machine learning techniques and software defined networking and, base paper. In chapter 3, analysis of normal traffic and DDoS traffic is discussed as discussed in base paper. In Chapter 4, we discussed the machine learning based DDoS classification system and its architecture. In chapter 5, Results are discussed and compare these results with base paper. In chapter 6, conclusion and future work are discussed.

Chapter 2

Literature Review

In this chapter, we discussed different techniques to detect, mitigate and prevent DDoS attack in a cloud computing environment. It is important to know the difference between legitimate packets [21] and illegal packets to tackle DDoS attacks in a cloud environment. Defensive techniques against DDoS attack are broken down into two categories. One category of those techniques which detect only DDoS attacks and another category of those techniques which not only detect but also include DDoS prevention. We have considered research articles of the past 6 years because recent arising problems and their solutions are addressed in it. There are 34 surveyed papers in which DDoS detection system is discussed and 14 surveyed papers in which the DDoS detection techniques and DDoS prevention techniques are discussed. The first research question [1] which is raised in chapter 1 is answered in 2.1 and 2.2.

2.1 DDoS Detection Techniques

To detect DDoS attack in cloud environment, some features are required. Selection of these features is the initial phase and using less number of features reduce computational resources, time complexity and provides a cost effective solution.

[1] What are the limitations in surveyed machine learning techniques?

Osanaiye et al. proposed Ensemble-based multi-feature filter selection method (EMFFS) [2] that is a mix of information gain performance, gain ratio, chi squared and Relief. This technique provides high accuracy and detection of DDoS having 13 features instead of 41. These features produced 99% accuracy with low rate of false detection. These selected features to be tested on other machine learning classifier to test attribute selection technique.

An efficient DDoS defense mechanism is based on these three characteristics. It is capable of preventing, detection and mitigate when DDoS attack occur. Somani et al. did a lot of contribution to classify and discuses attack models [22] under classification. Factors have been discussed for building an optimistic and effective system. Author frightened to DeNy DDoS attack as if we apply current defense mechanisms on it would produce a huge amount of false alerts.

Attackers compromise the availability of virtual machines in a cloud environment by sending a lot of traffic to specific targeted VMs. To overcome this problem, an effective load balancing mechanism is required. Wahab et al. proposed two fold solution [3] that make confident between VMs through Bayesian inference and maximum game which is an efficient detection way by creating faith relation between hypervisor and attacker. This method produces 26% detection accuracy.

A machine learning technique discussed by He et al. which is based on statistical information and provides detection on the source side [11]. Attacks on SSH brute force, waves of ICMP, DNS reflection, and TCP attacks on SYN features have been selected and conducted an experiment. Results show 99.7% accuracy in detection of these selected features with few false alarms.

Pandey et al. proposed a statistical and distributed network packet filtering model [23]. For the detection of cloud based DDOS attacks. Basically, multiple packet filters are needed to be distributable between individual virtual machines that generate and share a normal behavior profile in a consistent interval with the coordinating node. Network attributes selected attributes are identical to the standard computational profile. Based on the usual actions, a decision is taken to accept or reject the incoming packet.

Ramakrishna et al. have developed a new form of DDoS attack called "EDoS" [24]. The purpose of this EDoS attack is to will the economic loss of the intended legal customer. The technique offers a two-sided defensive system against DDoS and EDoS threats. One factor is inbound and outbound traffic screening, where packets for legal use can be accessed. Another factor is keeping a white list of approved customers who can conveniently use cloud providers. This approach is highly successful against DDoS attacks in narrow networks.

Agarwal and Tapaswi et al. addressed a comparative study of low-level DDoS attacks [5]. They addressed all potential DDoS threats, the effects, strengths and limitations of previously developed identification, avoidance and mitigating strategies.

Chen et al. describes the characteristics of the controller vulnerable to DDoS attacks in dynamically defined networks [17]. They used four names for simulation software such as Mininet for emulating topology, Pox for managing, Hyenae for launching DDoS attacks and TcpDump for effective traffic processing. The XGBoost algorithm is based on malicious traffic analysis, provides parallel computation on a multicore computer, and constructs data matrix for data processing.

Auto-scaling systems help to shield the cloud infrastructure from distributed denial of service attack (DDoS) by adding devices. Bremler et al. addressed a new type of DDoS attack called "Yo-Yo" attack [25] in which the attacker sends traffic loads over a normal time span to slow down the scaling process. Scale up at the beginning and limit the inclusion of machines along with a traffic filtering may be a successful protective measure against Yo-Yo attack.

Wani et al. present a machine learning technique that is a combination of Support Vector Machine, Nave Bayes and Random Forest algorithms [14]. This hybrid SVM technique demonstrated outstanding performance and could be used for intrusion detection purposes. Support Vector Machine, Nave Bayes, and Random Forest score accuracy of 99.7%, 97.6% and 98.0% respectively. This technique could be testing by selecting the number of classifiers.

Aborujilah and Musa proposed a covariance matrix approach to detect HTTP DDoS attacks in a cloud environment [26]. Multivariate Similarity Analysis-Based Monitoring Methodology (MADM) analyses traffic activity to classify flood attacks that differ from regular traffic.

Gupta and Badve suggested a strategy that is helpful in detecting a DDoS threat. It is based on the Generalized Autoregressive Conditional Heteroscedasticity (GARCH) [27] model for forecasting traffic and artificial neural network (ANN) for filtering network traffic. The threshold has been raised where there is an unexpected shift in traffic, and the attack traffic is refused.

Idhammad and Belouch proposed an HTTP DDoS attack detection system in a cloud setting focused on the Knowledge Theoretic Entropy and Random Forest Ensemble learning algorithm [28]. The calculation of entropy, the pre-processing of data and the classification of traffic networks are three major steps. The network header entropy of incoming network traffic is determined using time-based sliding window algorithm.

Bhardwaj et al. classified various types of DDoS attacks under these three categories; Network or Volumetric DDoS attacks, DDoS attacks and TCP State-Exhaustion attacks [29]. Methods for protecting cloud infrastructure against DDoS attack is addressed on premise based DDoS approach, ISP DDoS solutions, Scrubbing Protection DDoS Prevention, and Multi-Tiered Network Architecture.

Bhushan and Gupta were hosting an assault on the word "Low-rate Denial of Service" [30]. It has low amount of traffic compared to DDoS attack and makes it difficult to detect as it appears to be normal traffic. A t-statistic-based hypothesis checking methodology can effectively diagnose LDoS.

In Software Specified Network and Cloud Environment [16] Dong and Abbas et al. addressed various forms of DDoS attacks. Application layer, Control layer, and Data layer DDoS attacks addressed in server infrastructure server attacks in SDN and botnet epidemic, large network access and resource roiling. This proposed techniques produced good results in detection of DDoS attack.

Li et al. spoke about LDoS in cloud-based container world. It's lightweight, and the atmosphere is quickly scaling-up. This statistical model based on queueing theory [31] in order to figure out system power and then to optimize the level of service QoS, dynamic approach is used to provide a customer with minimal system resources.

Jiao et al. proposed a strategy for detecting IP attacks from fixed sources (FSIP) and Random Source IP attacks (RSIP) [32]. It is a real-time TCP-based monitoring system that makes legitimate traffic and removes attack traffic by two step classifier.

Rukavitsyn et al. suggested a two-step self-learning method; to collect network traffic data through the Netflow Protocol and to relearn the detection model with new data [33]. The relearning algorithm is based upon the traffic threshold estimation. This method accurately detect DDoS attack.

Borisenko et al. have developed an algorithm for detecting internal and external DDoS attacks in the cloud environment [34]. It is based on a data mining monitor, traffic information storage controller and data mining analyzer, which warns firewall to prevent.

Sophia and Gandhi have suggested a Stealthy DDos Detection Mechanism [35]. When a request by a single user crosses the threshold, the IP address of that user is blocked. But the unethical person is not able to achieve any immoral operation.

Borah et al. also attempted to develop an intrusion detection algorithm introduced by Nadya et al. The suggested algorithm [36] enhancement is improved by applying the self-organizing map (SOM) to the training process. The methodology leads to the fact that, while SOM is qualified to detect unlabeled intrusions, it might not always be able to achieve positive outcomes. The SOM must be trained regularly using various nodes to achieve better results.

Hoz et al. developed a scheme for network intrusion detection [37] based on the self-organizing map (SOM) and principal component analysis (PCA). In addition, data set noise and low variance characteristics are filtered using PCA and FDR. Fisher

discriminant ratio (FDR) was considered as a feature selection and noise reduction feature, the purpose of probabilistic self-organizing maps (PSOM) is to model a feature space and allow to distinguish between regular and malicious connections. Using the estimation of the activation probabilities during the training process, sensitivity, precision and accuracy values up to 97%, 93% and 90% respectively. By varying the previous activation probabilities of the SOM modules, precision could be increased.

Yang et al. suggested a novel anomalous network traffic analysis technique to detect DDoS in the cloud computing environment. For identification purposes, these six characteristics were selected: the number of source IP addresses, the number of source port numbers, the number of destination IP addresses, the number of destination port numbers, the number of packet types, and the number of network packets. Proposed entropy of hybrid data and SVM model [10] to solve the problem of classification. Finally, experimental findings indicate that the proposed algorithm can detect highly reliable anomalous network traffic.

Popular problems for the IDSs are huge volumes of computing data, poor identification rates and high rates of false alarms. [12] is provided as a technique based on the Online Sequential Extreme Learning Machine (OS-ELM) for intrusion detection. Singh et al. also introduced a methodology that uses alpha-profiling to minimize time complexity, while redundant features are eliminated using a collection of Filtered, Correlation and Consistency-based feature selection techniques. Beta profiling is used instead of sampling to reduce the scale of the training dataset. NSL-KDD dataset is used for experiments. The proposed methodology tackles numerous problems related to the IDS and the network traffic dataset.

In the anomaly detection process, the detector uses network traffic statistics, such as the incoming packet header field entropy (e.g. source IP addresses or protocol type). Calculates the quantitative attribute detected and causes a warning if an extreme variance happens. They can be quickly compromised by spoofing attacks. Elik and Brooks clarified the weakness of entropy-based network monitoring schemes [38] and proved with an own generated dataset.

Cloud deals with a large volume of data and requires the Intrusion Detection System (IDS). Reasonable preparation is needed to detect all intrusions immediately and accurately. The presence of a trivial feature set in training data increases memory space and training time. Ghosh et at. developed feature selection [39] through a CS-PSO algorithm over NSL-KDD dataset. Feature Collection from a high-dimensional dataset is a safer solution to training time and memory storage that the CS-PSO algorithm demonstrates by improving the IDS capability.

According to the 2013 Prolexic Quarterly Global DDoS Attack survey, the overall number of attacks grew by 21.75 percent compared to the same quarter in 2012. Ozcelik et al. also suggested a novel approach [40] to detecting DDoS: Cusum-Entropy. Detected threats with high identification and low false positive rates. The entropy of the source IP address used in this analysis, but this method can also be extended to the entropy of other packet header fields.

In order to distinguish both low-rate and high-rate DDoS attacks, Bhuyan et al. benchmarked major data metrics such as Hartley entropy, Shannon entropy, Renyi's entropy, generalized entropy, Kullback-Leibler divergence and generalized information distance measure [41]. They found that using an acceptable data metric helps to magnify the difference between legitimate and attack traffic in real-world network traffic. Low overhead computation is another important benefit of such a metric in the real-time monitoring of DDoS threats.

Unlike typical data flow-based detections, which are highly focused on data flow patterns, Cao et al. suggested an approach that would take advantage of the virtual machine state, including CPU use and network use [42], to classify the threat. They note that when an attack is initiated, malicious virtual machines, display identical status trends. Unreasonable allotment of network resources contributes to this kind of intrusion, if we should reallocate network resources according to the VM's running pattern, this kind of DoS attack could be removed.

A mixed methodology for achieving a high identification rate with a low false positive rate is mentioned. Guo et al. proposed a two-level hybrid approach [43], consisting of two anomaly - based components and a misuse - based component.

In stage 1, an anomaly detection system with low computational complexity is built and used to construct a detection component. The K-nearest neighbor algorithm is becoming critical in building the two different identical components for 2nd stage. The experiment conducted on the KDD'99 dataset reveals that the proposed hybrid solution is capable of detecting known and unknown attacks. It detects network irregularities with low false positive rate and high detection rate, indicating that it is a successful candidate for intrusion detection.

In order to test the efficiency of the dynamic entropy model, Jian et al. contrasted it with the standard information entropy model. With regard to the three features of live communication, encounters, inherent trends and creativity, a communication system model consisting of a controlled network and an external environment was developed. The experiment shows that the treatment of anomalies in this way provides effective and higher expertise in the identification of anomalies. The dynamic entropy-based approach was found to be more adaptive to [44] and capable of capturing anomalies.

Lee et al. proposed a technique for constructive monitoring of DDoS attacks by leveraging its architecture, which consists of the collection of handlers and agents, coordination. 2000 DARPA Intrusion detection scenario specific dataSet was used. The dataset is categorized into five distinct phases, such as the normal, phase 1, phase 2, attack and post-attack groups, respectively. Among the five phases of the DDoS attack, the proposed approach detected three phases efficiently [45] and indicate that each step of the attack situation is well subdivided.

To detect DDoS in the cloud environment, Signature based detection system, Anomaly based detection system, Software Specified Networking (SDN), Machine learning algorithms and hybrid techniques were addressed. Signature based detection is much faster [7] among all other techniques, but has a low rate attack detection. Machines learning techniques are producing better results with high detection rate with minimum false alerts.

Detection technique only detects the malicious activity inside a network. These techniques only alerts when a DDoS is detected. After surveying of these detection

techniques, machine learning detection techniques are more suitable and fast, as compared to other discussed techniques because it is helpful in the detection of both know and unknown attack with minimum time and space complexity. But These systems do not provide any mechanism of mitigation or avoidance of DDoS attack.

2.2 DDoS Prevention Techniques

DDoS prevention techniques not only detects, mitigates and responds against DDoS attack. These techniques first identify the existence of an attack, finds the source of the attack and blocks it until a normal behavior is being observed. There are 14 different DDoS prevention techniques are discussed in this section.

Cloud computing is becoming popular in Enterprise IT environment as it is providing cost effective and scalability. Software Defined Networking (SDN) is providing efficiency in networking management, but using it in a cloud environment raised DDoS attack occurrence. Wang et al. proposed DaMask-D [46] which is a hybrid of anomaly based detection and attack mitigation modules. Previous graphical probabilistic inference models have testing and training phase, but DaMask-D added updating phase that helps in solving dataset shift problem.

Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks are affecting the company's name and economic loss by compromising the availability of the system. Gupta and Badve [21] discussed Cloud environment, multiple forms of DDoS attacks. A lot security challenges are there in cloud computing that solves the availability, confidentiality and integrity of the system. Volume attacks, protocol attacks and application layer attacks are common types of DDoS attack. The attackers main goal is to get financial gain by alarming or steal data by distracting. For detecting DDoS attacks and DoS attacks, several network monitoring tools are available. Packets filtering, malicious traffic identification, systems training is the big challenges in a cloud environment to defend against DDoS attack. The challenges could be explored by researches in future.

DDoS mitigation service performance relies on its quality of service not affected and post attack time until normal condition. Somani et al. presented a framework that is the hybrid of affinity-based victim-service resizing algorithm (Resource Shrinking and Expanding) and TCP tuning technique [6] targets to provide quick and sustainability where quick targets is to reduce mitigation time and sustainability targets cost effectiveness by mitigation in available resource.

In this paper [47] Zarepoor et al. explained a model for the detection and mitigation of the DDOS cloud computing-based attacks. This model required a limited storage and a rapid detection ability. Experimental findings show that most DDOS attacks can be detected by the system. The performance of the proposed system was evaluated using metrics. A high detection accuracy (97%) was achieved with low false.

Devi et al. suggested a mathematical model based on chi-square statistics for the identification of DDoS attacks in the cloud environment [48]. This model consists of the construction of cloud test beds, the production of benchmark profiles, cloud output monitoring, the identification of DDoS and the defence mechanism to block malicious sources.

R.Kesavamoorthy et al. researcher focused on method to detect and prevent from DDOS [49].Agents used particle swarm optimization for strong communication between themselves and also for making decisions. By using multiple agents attacks are detected. Agents communicate with each other and update the coordinator agent. Coordinator agent analyzes the current scenario. Monitoring agent keeps eye on the network and cloud resources. If any unusual thing happens, it activates detection and recovery agent to take action.

Bhuyan et al. suggest E-LDAT, a lightweight extended-entropy metric method [50] for both DDoS flood attack prevention and IP (Internet Protocol) trackback. Their target is to accurately classify DDoS attacks by calculating the metric gap between legal traffic and attack traffic. An EEM-based IP takeback scheme was suggested and able to trackback zombies traffic. This traffic sends a stream of traffic to the cloud side to disturb its quality of services.

Alsirhani et al. explain a DDOS detection technique for cloud computing resources. Their anticipated system comprises of three steps; Classification algorithm, Parallelism computing and Fuzzy Logic system [1]. The proposed framework uses a classification algorithm to identify and avoid DDOS attacks on packets of traffic. The principle of parallelism is used to speed up the execution of the classification algorithm. The fuzzy logic makes the selection of the following classification algorithm. A testbed is configured for the evaluation of the classification algorithm and DDOS detection parallel calculation. MATLAB tool is used for testing fuzzy logic system.

Software Defined Networking (SDN) is an powerful strategy for tracking and managing Internet traffic and can easily prevent DDoS attacks. Bushan and Gupta address the core features of the Software Defined Networking [15] for use in a cloud environment against a DDoS threat. They build a flow-table space for a switch that depends on the idea of a queuing theory. This strategy uses the unused flow table of other OpenFlow switches in the network to shield the switch table from overload.

Agarwal and Tapaswi addressed defence strategies against various forms of distributed denial of service (DDoS) attacks [7]. They have been listed under Low Scale, Conventional and Economic Rejection of Sustainability Assaults.

Jabel et al. have developed a model to prevent distributed denial-of - service attacks that depend on intrusion detection methods focused on host over hypervisor environments [51]. This model is based on three components; the combination of main component and linear discriminant analysis, the "Ant Lion optimization" algorithm for feature selection and the cloud configuration of artificial neural networks. In host-based IDPSs method, there are two phases, IDS blocks malicious TCP and UDP traffic and IPS blocks malicious IPs and table updates.

Tsai et al. have combined virtualization technologies with the concept of depth security based on the user defined network (SDN) [13]. The SDN controller program tracks incoming packets and blocks malicious ports. The Network Intrusion Prevention System further investigates and prevents suspicious data for the future.

Pillutla and Arjunan present a Fuzzy self-organizing DDOS-based (FSOMDM) mitigation technique [52] that effectively replaces the neurons of the standard kohonen neural network model by updating the Fuzzy Rules. These rules track and identify network traffic as malicious or natural.

DDoS security strategies not only detect malicious traffic, but also blacklist malicious traffic. Software Specified Networking (SDN) and Machine Learning techniques have been used not only to detect, but also to avoid against DDoS attacks. Policies are developed in the system to find a source of traffic and to drop a request from a malicious source. The malicious source is not excluded from the blacklist until its normal activity has been detected.

Saxena and Dey suggested a third party auditor (TPA) based packet trackback technique called "Internet Warrior" [53] that uses the Weibull distribution to locate the root of the DDoS attack. It also alerts the cloud under the DDoS attack. They addressed the "IP-spoofing" issue. This technique defends agaist DDoS with less time complexity.

Detection and prevention techniques not only detects the malicious activity inside a network, but also prevent a system from DDoS attacks. After surveying of these detection techniques, machine learning detection techniques are more suitable techniques because these techniques pick the most relevant data attributes along with different classifiers to achieve improve results. Techniques other than machine learning, are not producing good results in a dynamic environment. Using a minimum number of attributes for classification, time and space complexity remains low.

2.3 Comparative analysis of surveyed techniques

Different approaches discussed in literature to detect and mitigate DDoS attack in a cloud computing environment. In these techniques various datasets used to perform experiments. Similarly, various detection and prevention technique have

been suggested which produce effective results against DDoS attack in a cloud computing environment. But some limitations also have been discussed in these techniques. Tables are created on the basis of the techniques they used. There are two parameters discussed in each table. One is the reference of the paper and summary. In summary, working of the proposed technique, the data collection used and the drawbacks are addressed. The following are the techniques discussed in the tables 2.4,2.1 and 2.2 below.

1. Entropy Information
2. Machine Learning Techniques
3. Software Defined Networking (SDN)

Comparative analysis of surveyed techniques in which machine learning methods and classifiers are discussed, given in Table 2.1. There are three parameters discussed in the table. Reference of the paper, which methodology is used to detect, mitigate and prevent from DDoS attack and summary. In summary, working of the proposed technique, the data collection used and the drawbacks are addressed.

TABLE 2.1: Comparative Analysis of Surveyed Machine Learning techniques to Defend DDoS Attack

References	**Technique**	**Summary**
Wang et al. [46]	Anomoly based detection and prevention technique	Labelled data is used for defense mechanism. The UNB ISCX dataset is used.
Osanaiye et al. [2]	Ensemble-based multi-feature selection method	High accuracy and detection of DDoS attack. 13/41 features (attributes). NSL-KDD dataset is used.
Wahab et al. [3]	Hybrid of Bayesian and Maxima game classifiers	Effeciently detect by building a bond of trust between the hypervisor and the intruder. Test traffic generated in laboratory.

He et al. [11]	Hybrid classifiers	Efficiently detect attack with minimum false alarm. Own generated dataset.
Zareapoor et al. [47]	DDoS detection and mitigation model	required a limited storage and a rapid detection ability. UCLA, DARPA and CAIDA datasets used because of nature of real time monitoring.
Kesavamoorthy et al. [49]	Swarm optimization	Coordination of multiple against produced attck detection with great accuracy.
Pandey et al. [23]	Packet filtering model. DARPA dataset used.	Master node accepts or reject request by coordination with agent nodes.
Alsirhani et al. [1]	Hybrid system of fuzzy algorithm, classifier and paralellism computing.	Random Forest output most efficient results as compared to Nave Bayes, DT(Entropy), DT(Gini) and Random Forest . Dataset is produced through MATLAB.
Ramakrishna et al. [24]	Inbound and outbound traffic screening	Proposed technique is baed on two-sided defensive system against DDoS and EDoS threats. Dataset is generated in lab. Only efficient in narrow networks.
Chen et al. [17]	Hybrid of different algorithms	XGBoost, parallel computation and data matrix used for detection. KDD 99 dataset used. Old dataset is used for testing.
Bremler-Barr et al. [25]	Scalling configuration	Scalling up nd down help to detect attacks. Limited resources produced inefficient detection.

Wang et al. [46]	Graphical model-based detection module	The UNB ISCX dataset labeled the DDoS attack network traffic produce accurate detection.
Gupta et al. [27]	GARCH model and artificial neural network (ANN) classifier	GARCH model and artificial neural network (ANN) to classify traffic. KDD CUP 1999, NSL-KDD, DARPA 2000, and CIDD datasets used.
Idhammad et al. [28]	Theoretic Entropy and Random Forest Ensemble learning algorithm	Entropy, the pre-processing of data and the classification of traffic . Adopted CIDDS-001 dataset used. Only HTTP DDoS detection.
Jaber et al. [51]	Hybrid of Knowledge theoretic entropy and Random Forest classifier	Blocks malicious Block malicious TCP and UDP traffic. CIDA and UCLA datasets used. Limited to TCP and UDP flooding attacks.
Bhushan et al. [30]	T-statistic-based hypothesis method	DARPA dataset taken as attack-free and CAIDA dataset taken as attack traffic. Only detects low rate DDoS attacks.
Jiao et al. [32]	Features selection approach	Real time monitoring of attacks from fixed sources (FSIP) and Random Source IP attacks (RSIP). ISCX IDS and CAIDA datasets used.
Rukavitsyn et al. [33]	Self learning method	Classify traffic to distinguish between legitimate and malicious traffic. Dataset is produced in laboratory. Used static Machine Laerning techniques. Static Machine Learning techniques used.

Pillutla et al. [52]	Fuzzy self organizing maps-based DDoS mitigation mechanism (FSOMDM)	FSOMDM and neural network model efficiently produce accuracy.
Borisenko et al. [34]	Real Service in Virtual Network Framework (RSVNet)	Data mining techniques are used to classify internal and external attacks in cloud.
Devi et al. [48]	Chi-square statistics	Laboratory generated dataset simulate the traffic behaviour of ICMP , UDP , Land , TCP SYN and TCP SYN-ACK floods and efficitenly detect these attacks.

Comparative analysis of surveyed techniques in which software defined networking (SDN) technique is discussed is given in Table 2.2. Reference of the paper and summary are the used parameters. In summary, working of the proposed technique, the data collection used is given.

TABLE 2.2: Comparative Analysis of Surveyed Software Defined Networking Techniques

References	**Summary**
Dong et al. [16]	Classify the layering attacks. NSL-KDD dataset used.
Bhushan et al. [15]	Queuing theory shields the switch table from overload. NSL-KDD dataset is used.

Comparative analysis of surveyed techniques in which entropy selection techniques are discussed is given in Table 2.4.

TABLE 2.4: Comparative Analysis of Surveyed Entropy Selection Techniques

References	Summary
Cao et al. [42]	Entropic importance of network traffic and CPU use to identify a malicious request. Does not fit effectively with fewer VMs
Ozcelik et al. [40]	Alerts when a high deviation is seen on the basis of entropy. Test traffic generated in laboratory. Unable to detect spoofed attacks
Jian-Qi et al. [44]	Using entropy and seeks a traffic flow correlation to identify an attack. Test traffic generated in laboratory. Unable to detect spoofed attacks.
Yang et al. [10]	Used 6 criteria for network traffic. Entropy and SVM data is used to determine the request. DARPA, KDD-CUP 99 and NSL-KDD datasets used. Selection of less feature affect detection.
Jun et al. [29]	Detect using traffic volume and entropy data in packet header. OPNET simulattor is used for traffic generation. Live traffic not considered.
Bhuyan et al. [41]	Low and high-rate DDoS attack identification. CAIDA and TUIDS dataset used. Unable to detect low rate attack.
Bhuyan et al. [50]	The identification of DDoS and IP traceback is carried out using the expanded entropy scheme. MIT Lincoln laboratory tcpdump data, CAIDA, and TUIDSdatasets used. Hard to track low-speed attacks.
Lee et al. [45]	Entropy of source, destination IP addresses and source, destination ports are used to classify attacks. DARPA dataset is used. Low attack detection rate.
Ozcelik et al. [40]	Signal analysis with the estimation of the packet header attributes to identify an attack. Test traffic generated in laboratory. Live traffic not detected

2.4 Research Paper used as base for Proposed Technique

Machine learning based classification techniques produce better results in term of accurate and true detection of attack [18] to a defend DDoS attack in a cloud computing environment. Verma et al. presented a system which is based on machine learning techniques. This system is divided into three subsystems; pre-processing, adaptive attributes selection and detection and prevention. Formulas

used in different subsystems are mentioned in Chapter 4.

2.4.1 Preprocessing Subsystem

Preprocessing is based on two modules; Attributes extraction and Normalization. At the start, attributes are extracted from the incoming traffic. They used NSL-KDD dataset and attributes are extracted from this dataset. There are 41 features (attributes) present in this dataset with different data types. These different data type values needed to be normalized to bring them all on a standard scale [0-1]. 80 % of the data used for training and 20 % used for testing.

2.4.2 Adaptive Attributes Selection Subsystem

It is based on three modules; probability, entropy and threshold selection. At the start, the probability of all attributes acquiring the given anomaly is calculated through a general probability formula. NSL-KDD dataset is used for experiments. This dataset is based on 41 features (attributes) and a class (normal/anomaly) attribute. The entropy of Shannon is used to determine the divergence randomness for every attribute. Threshold selection task is the most [18] important task. DDoS attacks may be avoided by setting a proper threshold value. Most of the attribute selection approaches for incoming traffic classification are based on static threshold statistics. These static threshold values are unable to produce good results against DDoS detection. Therefore, a dynamic threshold selection approach is required to deal with different networks and incoming traffic conditions. There are four adaptive threshold values selection techniques that have been discussed. These are Interquartile range (IQR), Mean absolute deviation (MAD), Median absolute deviation (MedAD) and Bernsen. If the calculated entropy of is greater than threshold value then it is selected. Mean absolute deviation (MAD) is better as compared to other three adaptive threshold techniques. Median absolute deviation (MedAD) is also producing good results as compared to Mead Absolute Deviation (MAD) for adaptive selection of features (attributes).

2.4.3 Detection and Prevention Subsystem

For detection and prevention, six classifiers [19], [20], [18] called Decision Tree (DT), AdaBoost (AB), and Support Vector Machine (SVM), K-nearest neighbor (KNN), Random Forest (RF) and Multilayer Perceptrons (MLP) are used against each selection technique. Results show that the proposed approach with MAD thresholding technique and random forest classifier gives the better results. The accuracy of 98.226%, the detection rate of 98.066% is achieved with a proposed approach (MAD-RF). In the future, the detection accuracy for UDP and ICMP-based DDoS attack in the cloud environment can be improved.

2.5 Summary

In this chapter, we discussed different surveyed techniques to detect, mitigate and prevent the cloud environment from DDoS attacks. Techniques which only provide detection of DDoS attack was discussed in 1st subsection and Techniques which provides detection as well as prevention was discussed in 2nd subsection. In last subsection, comparative analysis of different surveyed techniques which were discussed in literature is listed. It was discussed under three different techniques named as entropy selection, machine learning and software defined networking (SDN). The base paper approach is discussed in detail in the last section. This will help in the development of our experimental system configurations and results comparison.

Chapter 3

Analysis of DDoS Attack

In this chapter, Analysis of normal traffic and DDoS attack is discussed. This analysis task is based on published research paper [18]. In the first section, system configurations and analysis of DDoD attack is presented as discussed in base paper. This chapter is further divided into proposed system configurations, normal traffic analysis, DDoS traffic analysis and summary. We analyze the behavior of these types of TCP, UDP and ICMP traffic under both normal and attack scenarios. The behavior each traffic type is analyzed through I/O graph. The virtual environment is built on laptop.

3.1 System Configurations and Traffic Analysis discussed in Publised Research Paper

To study the behavior of incoming traffic, a system is required to perform experiments. An experimental setup is built in a research lab to analyze DDoS behavior analysis. A Dell physical model device with i7-4790 processor [18] works@3,60 GHz is used to build a virtualized environment. It is a quad core with 8 GB of RAM. They used 1 server machine that hosts an application on the Apache Web server and different numbers of malicious machines to examine DDoS behavior. Traffic analysis was also discussed under normal scenario.

Traffic behavior of TCP, UDP and ICMP packets is observed in both normal and attack scenarios. Under the normal scenario, TCP, UDP and ICMP packets are analyzed for 40 seconds and found to be 100% efficient in delivering these packets.

TCP, UDP and ICMP flooding attack traffic are analyzed. The server was unable to manage huge volume of packets after the attack was initiated because it is unable to handle huge amount of requests at a time. After sometime, server is no more able to handle the requests either from the benign client or from the attacker. Further, they sent ping packets to the victim computer with different frequencies (30, 50 and 70). It is observed that the packets loss rate is increasing when the frequency of packets is increasing.

3.2 System Configurations to Analyze System Under Normal and DDoS Traffic

Our system configurations are based on previously discussed system configurations and experimental setup. A virtual cloud system is built on a laptop. There are three different machines used; one is the host machine, second is Guest 1 machine on which a simple web application is hosted and third is Guest 2 machine to launch ICMP flooding attack. The Haier y11c host computer with an Intel CoreTM M-7Y30 processor operating at 1.61 GHz frequency is used. The Microsoft Windows 10 x64 operating system is installed with 8 GB of RAM. Virtual Box 6.1 is installed on the host machine and two virtual machines are installed on Virtual Box. The Microsoft Server 2017 based operating system is installed with 4 GB of RAM and dual cores on the Guest 1 virtual machine. The Kali Linux operating system is installed with 1.5 GB of RAM and single core on the Guest 2 virtual machine. The High Orbit Ion Cannon (HOIC) and Low Orbit Ion Cannon (LOIC) tools operate on the host machine to create traffic for both TCP and UDP flood attacks. HOIC and LOIC are open source network stress testing and denial of service attack programs designed to simultaneously attack. The tool, called hping3 is running Guest 2, which is virtually installed to generate traffic for an ICMP flood attack.

It is a TCP/IP packet assembler / analyzer with command line environment. The implementation of these tools will be discussed scenario under attack. Guest 1 is a victim computer that analyzes traffic and system activity in both regular and attack situations. Wireshark is installed on this machine to evaluate and display these statistics using IO graphs. System architecture is shown below in figure 3.1

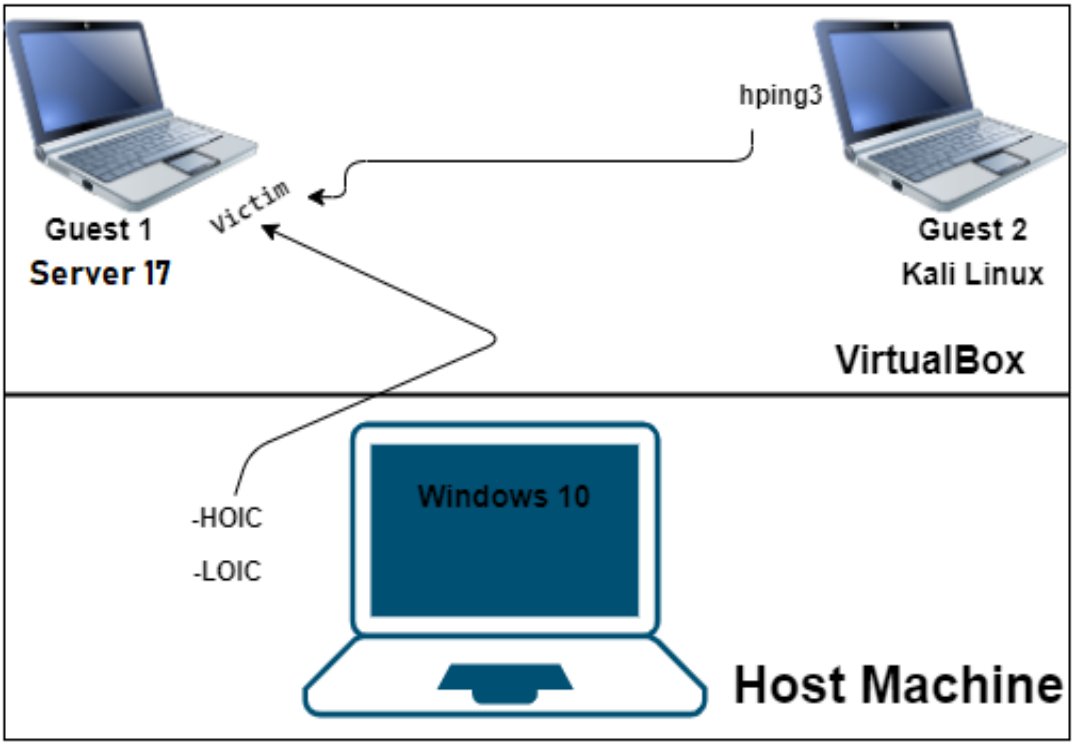

FIGURE 3.1: System Architecture for Analysis of DDoS Attack

3.3 Analysis of Normal Traffic

There is a normal stream of client packets to the victim system (Guest1). Traffic movement is tracked by a Wireshark tool installed on Guest 1. Various packets were sent to the victim computer (192.168.1.4) to track the performance of the device under normal condition. It is found that under normal circumstances the failure of the packet is 0 % due to 100 % delivery of packets. TCP, UDP and ICMP are the main types of packets sent to the victim system. We also evaluate the behavior of each type of packet as used as valid request packets and as attack packets.

3.3.1 Analysis of TCP Traffic under Normal Scenario

Transmission Control Protocol (TCP) is a connection-oriented communication protocol that allows the exchanging of messages between computer devices on the network. It is the most common protocol in networks that use the Internet Protocol (IP), also referred to as TCP/IP. TCP traffic is analyzed through IO graph under normal condition for 25 seconds. On the X-axis, number of packets are shown and on Y-axis, time period in seconds is shown. Maximum number of incoming and outflow of these packets is 12 packets per second is shown in figure 3.2.

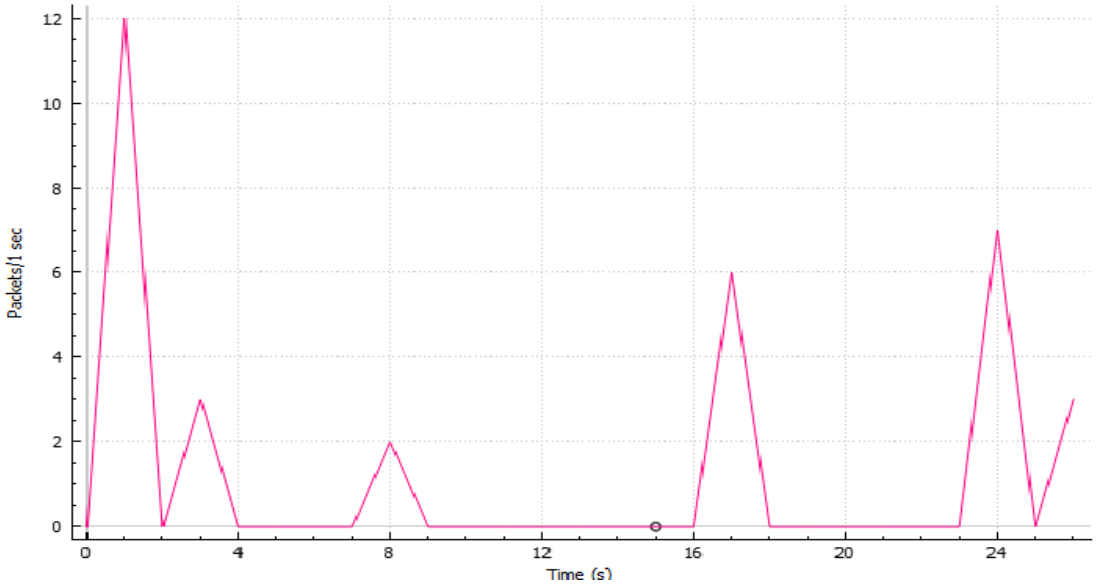

FIGURE 3.2: Visualization of Normal TCP Traffic through IO Graph

3.3.2 Analysis of UDP Traffic under Normal Scenario

The User Datagram Protocol (UDP) operates at the top of the Internet Protocol (IP) to transfer datagrams over a network. UDP does not enable the source and destination to set up a three-way handshake until the transmission takes place. It speeds up communications by not formally forming a connection until data is transmitted. UDP traffic is analyzed through IO graph under normal condition for 51 seconds. On the X-axis, number of packets are shown and on the Y-axis, time period in seconds is shown. Maximum number of incoming and outflow of these packets is 40 packets per second is shown in figure. 3.3.

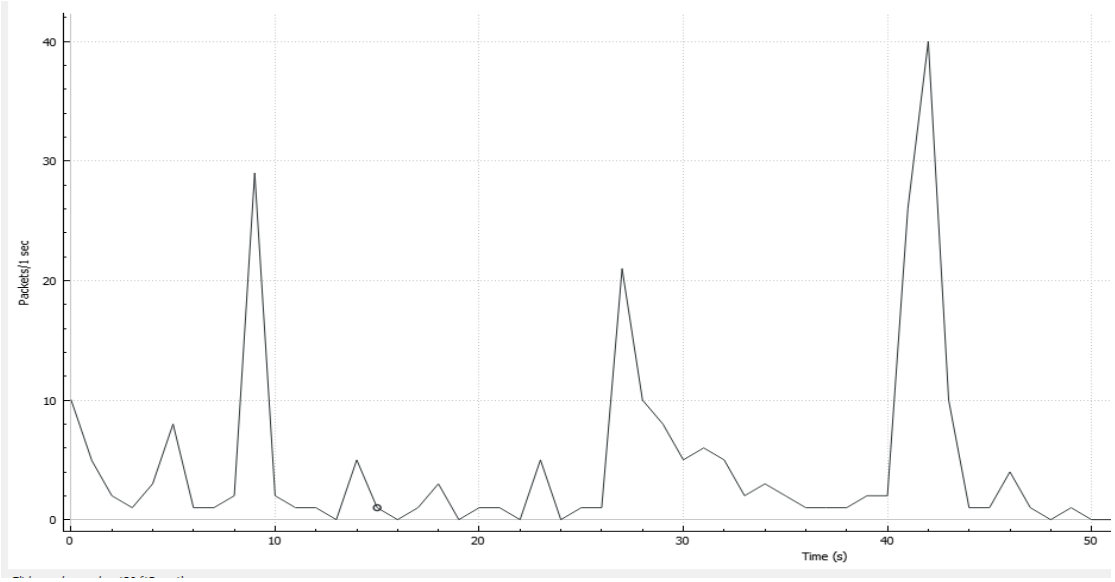

FIGURE 3.3: Visualization of Normal UDP Traffic Through IO Graph

3.3.3 Analysis of ICMP Traffic under Normal Scenario

ICMP is a network layer protocol used by network devices to diagnose network connectivity problems. ICMP traffic is analyzed through IO graph under normal condition for 51 seconds. On the X-axis, number of packets are shown and on the Y-axis, time period in seconds is shown. Maximum number of incoming and outflow of these packets is 22 packets per second is shown in figure. 3.4.

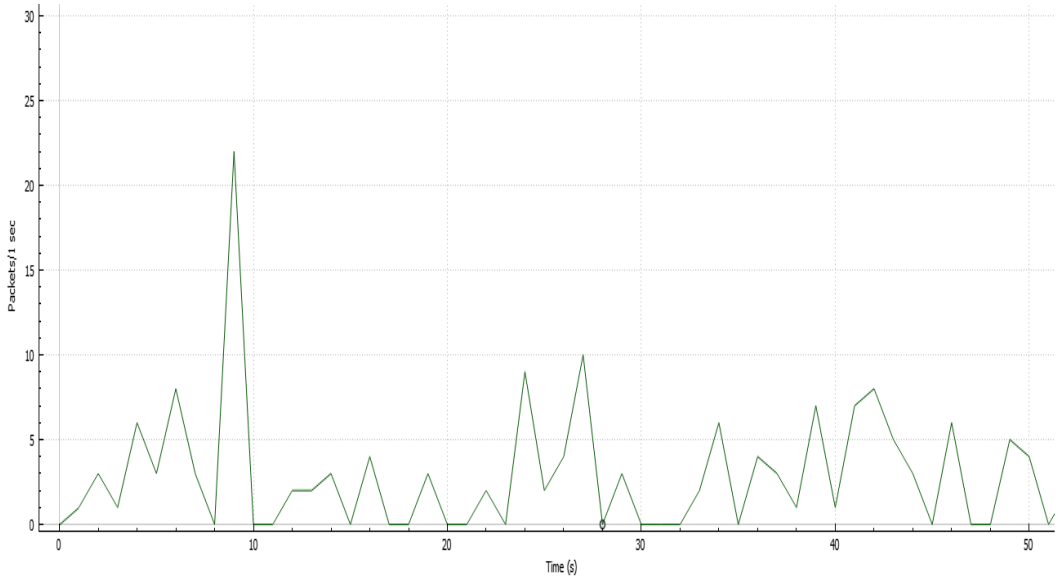

FIGURE 3.4: Visualization of Normal ICMP Traffic through IO Graph

Under normal scenario system resources are analyzed through "Resource Monitor". Usage of CPU and RAM are under 25% and there is few KBytes incoming and outgoing flow of packets. TCP, UDP and ICMP are the key types of packets sent to the network. We also evaluate the behavior of each type of packet as used as valid request packets and as attack packets.

3.4 Analysis of DDoS Traffic

In this analysis, the TCP, UDP and ICMP protocol-based floods were used to launch the attack. These flood attacks were produced using HOIC, LOIC and hping3. Guest 1 was unable to manage the stream of malicious packets when the attack was initiated because it had a small ability to handle the incoming traffic request at a time.

3.4.1 TCP SYN Flooding Attack

TCP SYN flood is a form of Distributed Denial of Service (DDoS) attack that exploits part of the usual three-way TCP handshake to overload the targeted clouds resources and make it unresponsive to its user and compromise its availability [54]. A normal TCP three-way handshake is established when a user sends SYN (synchronize) messages to the Guest 1 machine. This machine acknowledges these requests by sending SYN-ACK (synchronize and acknowledge) messages back to the user and finally user sends ACK (acknowledge) to the Guest 1. But in case of malicious user, Guest 1 replies with SYN-ACK (synchronize and acknowledge) messages back to malicious user. If the IP address is spoofed, never gets the SYN-ACK. This malicious user does not send ACK (acknowledge) to the server.

The H.O.I.C tool can launch [54] up to 256 simultaneous sessions of attack at once, taking down an entire target system by sending a continuous stream of junk traffic until legitimate requests cannot be processed any more. It has been developed in a scripting system to enable thrusters to be deployed, scripts designed to thwart

DDoS countermeasures and the increase DoS output. Ability to individually accelerate attacks with three settings: Low, Medium, and High. It has option to select the number of threads in a continuous attack.

In our tested environment, we generated TCP traffic of 50 machines and analyzed the traffic behavior for 100 seconds. At 26 seconds, the highest TCP traffic rate is 3100 packets per second. As computers are removed one by one, the inflow and outflow of packets reduce. Visualization of TCP traffic under attack scenario is shown in figure 3.5

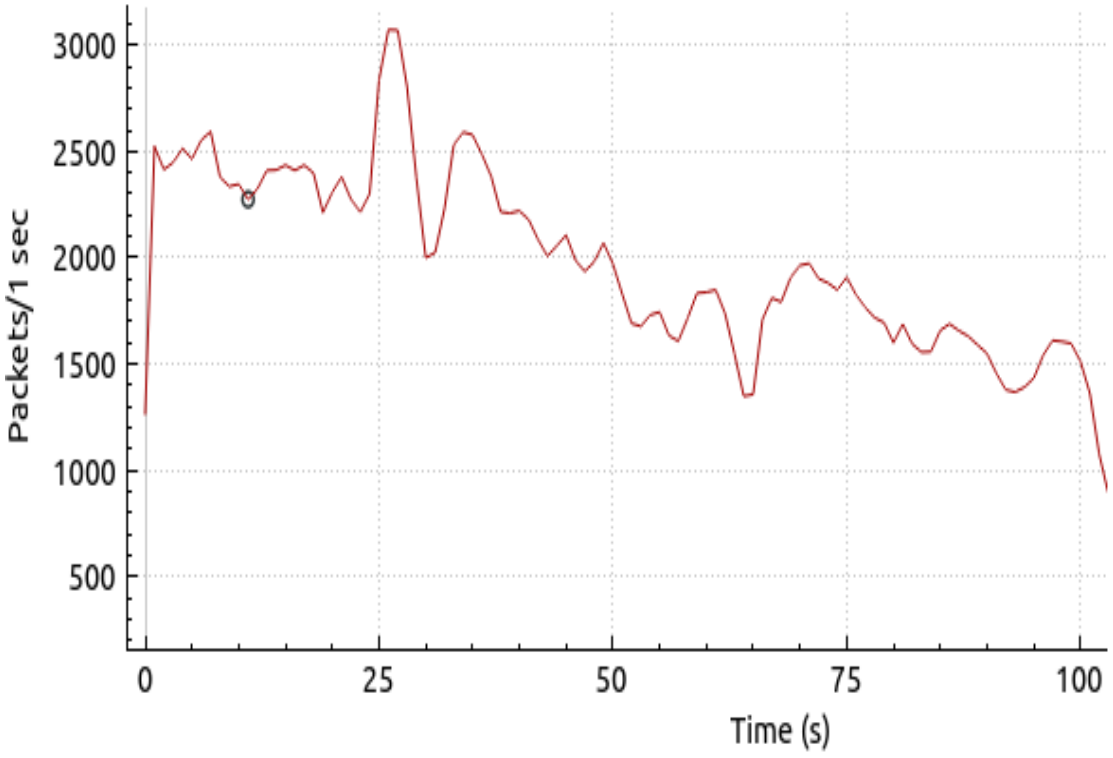

FIGURE 3.5: Visualization of TCP Traffic under Attack Scenario

3.4.2 UDP Flooding Attack

UDP is a connectionless data transportation system, and the attacker uses this connectionless mechanism to initiate the UDP flood attack. It is a type of denial-of-service attack in which attacker sends a large number of User Datagram Protocol (UDP) packets random (unknown) port on the server machine [Guest 1] and victim [55] respond with an ICMP (ping) packet to warn the sender that the destination

was unavailable. When the server machine (Guest 1) receives a flood of these UDP packets at a random (unknown) ports then it is unable to handle these requests.

L.O.I.C tool [54] is used to generate UDP flooding attack. It is known for being a very user-friendly and accessible tool and it gives users with very little technical skills the ability to launch DDoS attacks. Similarly HOIC, the target machine IP address is locked and can set the speed of the packet.

In our tested environment, we generated UDP traffic of 50 machines and analyzed the traffic behavior for 33 seconds. At 32 seconds, the highest UDP traffic rate is over 2500 packets per second. There is slight variations in traffic rate is observed due to changing of packet speed rate. Visualization of TCP traffic under attack scenario is shown in figure 3.6

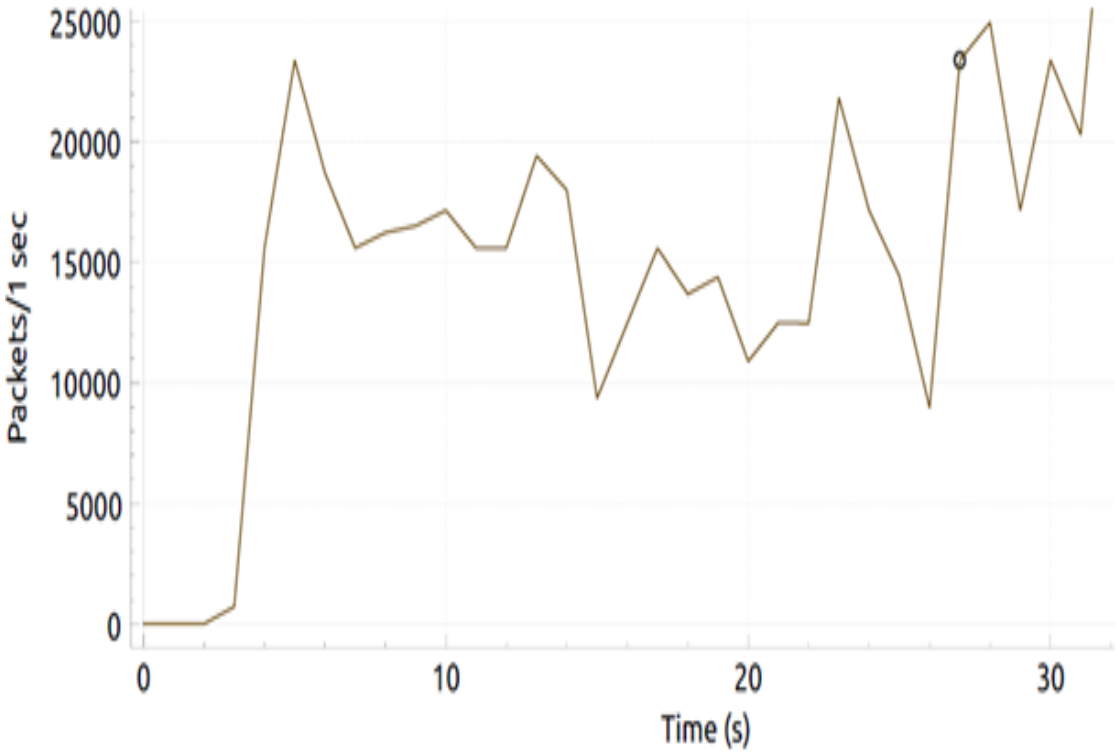

FIGURE 3.6: Visualization of UDP Traffic under Attack Scenario

3.4.3 ICMP Flooding Attack

An ICMP flood, also known as Ping, is a type denial of service attack in which the attacker tries to overwhelm a targeted device with ICMP echo-request packets

and make the target unavailable to normal users [56]. ICMP echo-request and echo-reply messages are usually used to ping a system on a network to test its health and connectivity. These requests are used to test the connectivity of two computers by calculating the round trip time from sending an ICMP echo request to receiving an ICMP echo response. However a malicious user sends multiple ICMP echo-requests with the help of different bots, the attack traffic is increased exponentially resulting in overwhelm a targeted system.

To generate an ICMP flooding attack, Hing3 tool is used. Hping3 is a TCP/IP packet assembler / analyzer [11] with the command line environment [61]. To generate traffic for UDP flooding attack, hping3 tool is run. This method helps you to monitor the size, volume and fragmentation of packets overwhelm the target and bypass or strike firewalls.

In our tested environment, we generated ICMP traffic of 50 machines and analyzed the traffic behavior for 33 seconds. There is some up and down in traffic rate is also observed. The highest ICMP traffic rate is over 3100 packets per second at 24 second. The traffic rate is consistent across 3000 packets per second. Visualization of TCP traffic under attack scenario is shown in figure 3.7

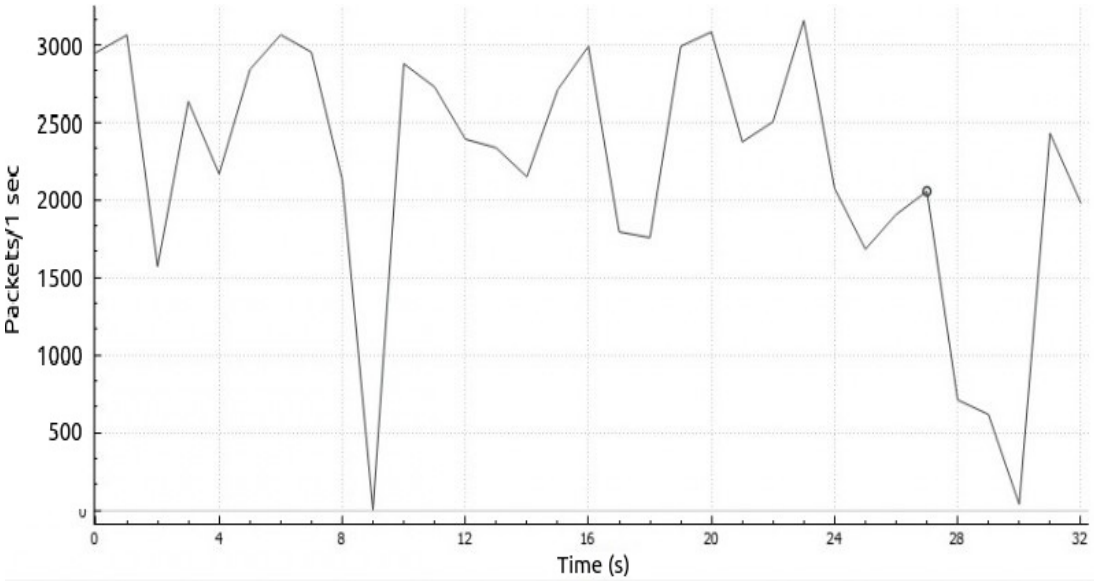

FIGURE 3.7: Visualization of ICMP Traffic under Atack Scenario

[11] http://www.firewall.cx/general-topics-reviews/network-protocol-analyzers/1224-performing-tcp-syn-flood-attack-and-detecting-it-with-wireshark.html

Under attack scenario system resources are analyzed through "Resource Monitor". Usage of CPU and RAM reaches up to 90% and there is huge increase in the incoming and outgoing flow of packets and packet loss rate reaches 0. It is also noted that the system is halted under these flood attacks. It is also noted that the system is hanging under these flood attacks.

3.5 Summary

System configurations, analysis of normal and DDoS traffic has been addressed in this chapter. In our experimental environment, there are two machines virtually installed on host machine by using Virtual Box 6.1. Guest 1 machine is used as victim machine and Microsoft Server 2017 is installed on it. HOIC and LOIC tool run on the host (malicious) machine to launch TCP and UDP flooding attacks respectively. Kali is installed on Guest 2 machine. It is also a malicious machine to launch an ICMP flooding attack using Hping3 tool. Further TCP, UDP and ICMP traffic are analyzed under both normal and attack scenarios. Usage of system resources is also observed under both scenarios. There is an exponential increase in usage of system's resources under different flooding attacks and rate of packet loss reaches 0.

Chapter 4

Proposed Machine Learning Based Classification of DDoS Attack

In this chapter, different machine learning based attributes selection and classification techniques are discussed. These techniques are reasonably capable of classifying the attack and benign requests and saving the cloud resources from the DDoS attack. At the start, we discussed the system architecture of already published technique [18]. This techniques has 3 main subsystems; preprocessing, adaptive attribute selection and, classification and prevention. Further, we discussed our proposed technique. Our proposed technique is capable of categorizing threats and normal requests and protects cloud storage from DDoS attacks. Our suggested technique consists of three modules: pre-processing subsystem, limited subsystem collection of attributes and subsystem identification and prevention. First defensive proposed system in the base paper is discussed and later we discuss our machine learning based DDoS detection system. Our Machine learning based classification of DDoS attack technique is based on discussed base paper technique. The research question [2] raised in chapter 1 is answered in this chapter.

[2]What is the minimum number of attributes which are used in our proposed machine learning based technique?

4.1 System Architecture of Published Research (Base) Paper

Cloud-based applications are growing day by day for a number of reasons due to their perpetuity and diverse dexterity. However, aggressive network traffic like Distributed Denial of Service (DDoS) plays a major role in challenging cloud-based applications. It is also important to protect against such attacks in order to conserve cloud capital. The defensive system is grouped into three further subsystems in base paper [18]. First subsystem preprocessing is further divided into two modules. One is the extraction of attributes from incoming traffic, but they used NSL-KDD dataset. These attributes are then normalized to bring all attributes on a standard scale value [0-1]. This data is then splited into 80% training data and 20% testing data. Training data is passed to the second module Adaptive attribute selection subsystem and testing data is passed to detection and prevention subsystem. Adaptive attribute selection subsystem is divided into three modules such as probability, entropy (Shannon) and threshold selection. Threshold calculating techniques are Interquartile range (IQR), Mean absolute deviation (MAD), Median absolute deviation (MedAD) and Bersen. If the calculated entropy value is greater that threshold, then it is forward to the next subsystem. In detection and prevention subsystem, these six classifiers named as random forest(RF), adaboost(AB) k-nearest neighboor(KNN), support vector machine(SVM), multilayer perceptrons(MLP) and decision tree(DT). These classifiers classify attack traffic and the normal traffic with high accuracy, low false detection rate. Results demonstrate that the proposed solution to the MAD thresholding technique. And the random forest grouping gives the best results. Separates the attack and benevolent request for the NSL-KDD dataset more specifically. MAD-RF is also capable of splitting DDoS attacks based on TCP, UDP and ICMP. The MAD-RF gives the sensitivity of 98.226 percent, the identification rate is 98.066 percent, the false warning rate is 0.019, the precision is 98.34, the AUC is 0.981 percent and the F1-score is 0.983, which is highest among all. System architecture of proposed system in base paper is shown in figure 4.1.

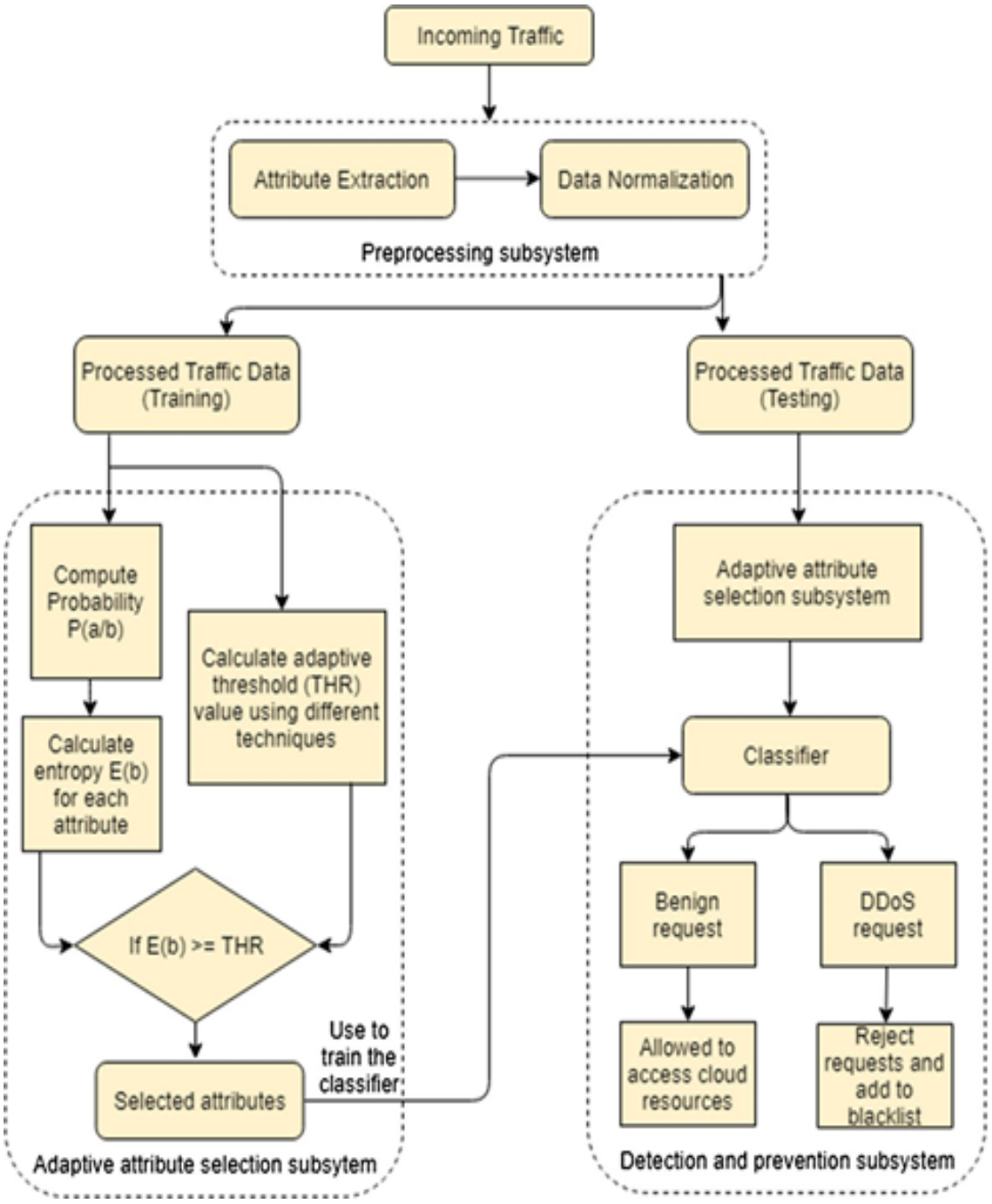

FIGURE 4.1: System Architecture Discussed in Base Paper[18]

4.2 Proposed System Architecture

The proposed system consists of three subsystems: preprocessing, adaptive attributes selection, detection and prevention. Attributes are collected and normalized from traffic in the Preprocessing subsystem at the beginning. Data is split

into subsets for training and testing. In the Adadtive subsystem, 80% data goes on and 20% data goes on for classification as partitioned in base paper[18]. Minimum numbers of attributes are chosen using different automatic threshold techniques in the Attribute Selection Subsystem. Finally, the detection and prevention subsystem is responsible for classifying traffic data as DDoS and normal. Fig 4.2 demonstrates the workflow of the proposed method.

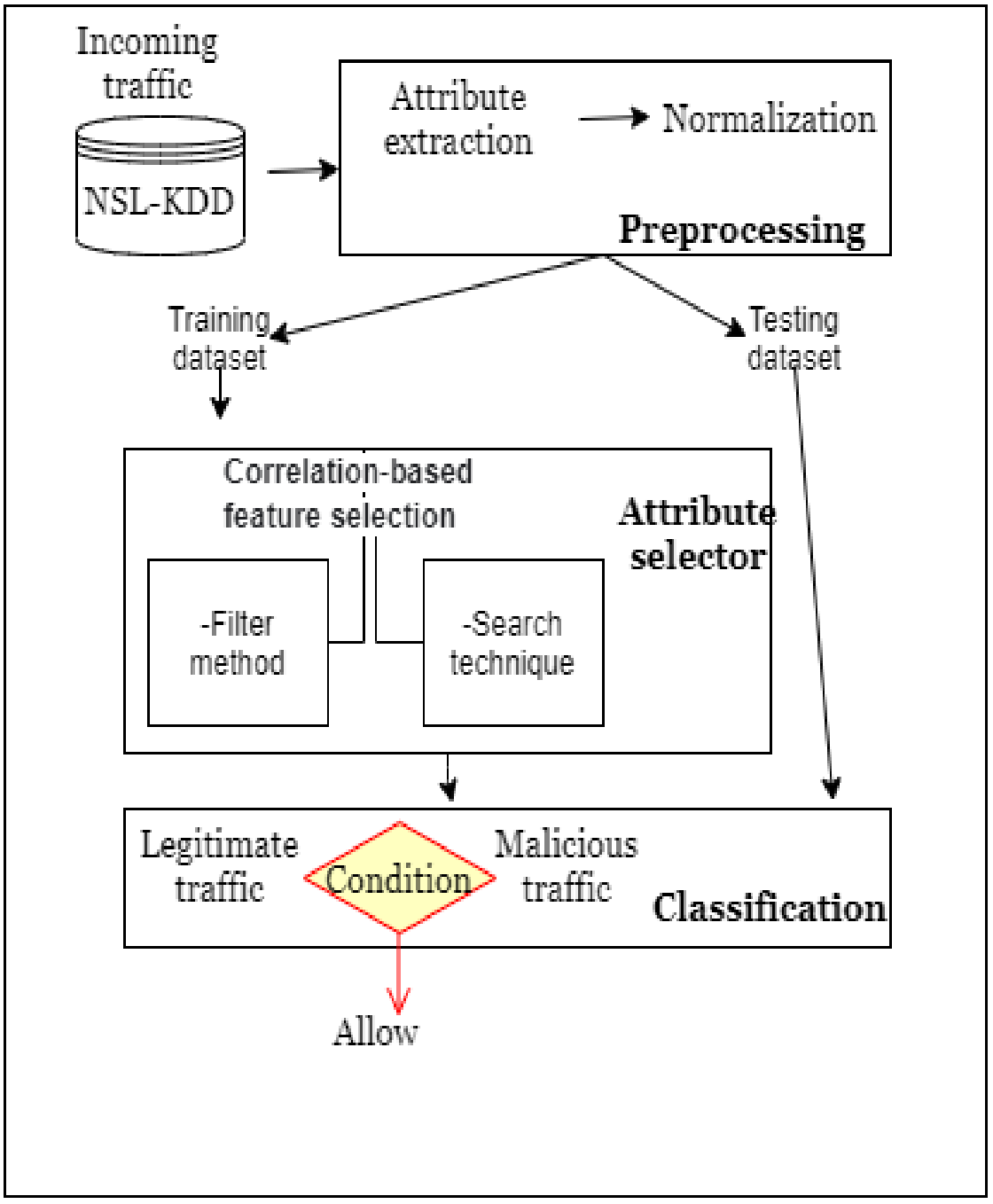

FIGURE 4.2: Proposed System Architecture

4.2.1 Preprocessing Subsystem

In the cloud system, incoming and outgoing packet information is contained in log files. Arrived packets contain both valid and malicious requests. These packets contain information, such as source_IP_address, destination_IP_address, logged_in, is_guest_login, and is_host_ login_source port, destination_port, flags, header_length, payload, class and much more. This knowledge is helpful in the drawing of attributes that aid in the identification of an attack. Therefore, the necessary attributes are extracted from the incoming cloud network in the pre-processing subsystem. However, a standard NSL-KDD data set is used. The comprehensive study of the dataset is discussed in Chapter 5. The attributes are derived from the this dataset and the values are standardized. Normalization is a scaling method used in the pre-processing phase.

4.2.2 Attribute Extraction

The attributes are derived from the incoming traffic. Packets provide various information about these attributes that contained in log files. Such features help to discriminate between legitimate traffic and malicious traffic. The NSL-KDD dataset [11] is used to determine the efficiency of the proposed machine learning technique. Same dataset is adapted in base paper [18]. It contains essential records of the complete KDD data set. There are a collection of downloadable files at the disposal for the researchers. KDDTest+.ARFF (Full NSL-KDD test set of ARFF binary labels) is used. It consists of 41 features (attributes) and a labelled class (anomaly or normal). There are four types of attack type: Dos, pobe, remote to local and user to root. Denial of service (DoS) is an attack category, which depletes the victims resources thereby making it unable to handle legitimate requests. We based on DoS in order to differentiate between legitimate traffic and malicious traffic. Features of this dataset are listed in table 4.1 These attributes are not set on a standard scale. These attributes are further passes to next phase for normalization.

[11] https://www.unb.ca/cic/datasets/nsl.html

TABLE 4.1: Features of NSL-KDD Dataset

S.No	Feature Name	S.No	Feature Name
1	Duration	22	Srv_diff_host_rate
2	Protocol type	23	Dst_host_srv_diff_host_rate
3	Service	24	Dst_host_serror_rate
4	Src_byte	25	Num_shells
5	Dst_byte	26	Srv_rerror_rate
6	Flag	27	Srv_serror_rate
7	Land	28	Rerror_rate
8	Wrong_fragment	29	Srv_count
9	Urgent	30	Same_srv_rate
10	Hot	31	Diff_srv_rate
11	Num_failed_logins	32	Is_guest_login
12	Num_compromised	33	Dst_host_same_srv_count
13	Logged_in	34	Count
14	Dst_host_diff_srv_count	35	Root_shell
15	Dst_host_count	36	Dst_host_srv_serror_rate
16	Su_attempted	37	Num_access_shells
17	Dst_host_same_src_port_rate	38	Dst_host_rerror_rate
18	Serror_rate	39	Num_outbound_cmds
19	Dst_host_srv_count	40	Dst_host_srv_rerror_rate
20	Num_file_creations	41	Is_hot_login
21	Num_root		

4.2.3 Normalization

Normalization is described as the method of transforming the original data without altering its behavior or existence. The features present in the dataset are of different data types and having different values. Therefore, it is needed to bring all values on a standard scale to apply machine techniques and classifiers. The purpose of the normalization is to adjust the values of the columns in the dataset

on a standard scale. The preprocessing module uses the minmax normalization methodology to bring all attributes to regular scale [0 - 1]. After normalization, the data is split into a training and testing datasets and passed to next subsystem for further process.

4.3 Attributes Selection Module

It is most important to use minimum number of features to achieve [55] maximum performance of the system. This helps in reduction of time and space complexity. In order to reduce the number of parameters, attribute selection module is used. Selection of attributes is a two-step operation, one is generation of subsets, and the other is ranking. Subset creation is a search method used to compare a candidate subset with already calculated subsets. If the new candidate subset returns better outcomes for any evaluation than the new subset is the good one. This process will proceed until the termination condition is met.

4.3.1 Search Techniques

The ranking of attributes is used to assess the value of the attributes. There are several rating approaches, most of which are focused on mathematical or information theory. There are two types of algorithms for collection of attributes. One is the Filter method, the other is the Wrapper approach. The wrapper approach 4.2 looks for an ideal subset of features customized to a specific algorithm and domain. The feature subsets chosen by the wrapper are smaller in size than the original subsets used by the learning algorithms. Attributes are measured on the basis of the evaluation metrics with respect to the characteristics of the data collection in the filter method.

In order to choose wrapper or filter method. We analyze these two approaches on the basis of their cost, computational time, scalability and attribute dependencies. A comparison table 4.2 is given below to discuss these parameters. Filter method

is choosen for our work because it takes a short time, simple and produce fast results. Because It is our intrest to select minimal number of attributes which are highly correlated to each other in minimum time.

TABLE 4.2: Comparison between Filter and Wrapper Approach [55]

	Filter	**Wrapper**
Computational time	Simple and fast	Complex and slower
Cost	Less expensive	more expensive
In terms of attribute dependencies	Only to some degree	Fully incorporated
Scaling ability to high dimensional dataset	Easy	Complex and slower

There are two ways of filtering techniques [56] called CFsSubsetEval and ConsistencySubsetEval, and two methods of search called GreedyStepwise and BestSearchFirst are used in our research. CfsSubsetEval (CSE) is the most better filtering technique [55] so that's why we used it in our work. CFsSubsetEval, together with BestSearchFirst and GreedyStepwise is producing maximum accurate results. Only 9 high correlated features (attributes) are selected while other (32) are not used for further processing. Merit of best subset is 0.435. The time complexity and quicker avoidance of DDoS in the cloud environment can be minimized by deleting other unnecessary (32) attributes.

4.3.2 Traffic Filtration Technique

We used the Correlation Feature Selection (CFS) Technique for traffic filtration purpose. Selection of features is a strategy for eliminating irrelevant and unnecessary features [57] that will help to increase the learning accuracy and classifiers' predictive accuracy. A subset function is useful if it is strongly correlated with the class, but not much correlated with other class features. CFS technique is based on the followings.

- Probability
- Entropy
- Information Gain

The probability of a feature is the either the existence of a favorable condition or not. This probability is used to find correlation between class attribute and the other attribute one by one. In this way, entropy of an attribute(x) is calculated that shows that how much these attributes are related with class attribute(x). A confidence matrix is a dynamically build on the basis of their relation. An Information Gain formula is applied in this data (subset of selected attributes), if these values are equal or higher than the confidence matrix value then it is picked as the best subsets among all other subsets.

Entropy-based knowledge theory [57] is used to find a connection between the class attribute and other attributes. Entropy is a measure of variance for a random variable. The following equation 4.1 can be used to find entropy.

$$H(X) = -\Sigma P(xi) log2(P(xi) \tag{4.1}$$

And the entropy of the attribute X (class) after observing the values of another element Y is described in equation 4.2.

$$H(X/Y) = -\Sigma P(yj) \Sigma P(xi/yj) log2(P(xi/yj)) \tag{4.2}$$

Here, P (xi) is the previous probabilities for all values of X, and P (xi/yj) is the posterior probabilities of X when values of Y are provided. The sum by which the entropy of X decreases reflects additional information on X (class attribute) generated by Y is referred to as information gain is given equation 4.3.

$$IG(X/Y) = H(X) - H(X/Y) \tag{4.3}$$

It can be inferred that feature Y is known to be more associated with feature X (class) than with feature Z, if IG(X/Y) >= IG (Z/Y).

In the most correlated attribute selection module, using above discussed CFS and BestSearchFirst techniques, the number of attributes decreased from 41 to 9 only. These attributes are most correlated to class attribute (normal/anomaly) are listed below.

- **src_bytes:** Number of data bytes transferred from source to destination in single connection [58].
- **dst_bytes:** Number of data bytes transferred from destination to source in single connection [58].
- **logged_in:** Login Status : 1 if successfully logged in; 0 otherwise [58].
- **serror_rate:** The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in srv_count [58].
- **srv_rerror_rate:** The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in srv_count [58].
- **diff_srv_rate:** The percentage of connections that were to different services, among the connections aggregated in count [58].
- **srv_diff_host_rate:** The percentage of connections that were to different destination machines among the connections aggregated in srv_count [58].
- **dst_host_srv_diff_host_rate:** The percentage of connections that were to different services, among the connections aggregated in dst_host_count [58].

- **dst_host_srv_error_rate:** The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_srv count [58].

4.4 Detection and Prevention Subsystem

In the detection and prevention subsystem, the test dataset will be used as input and delivered to the adaptive attribute selection subsystem to evaluate the optimum attribute collection. The classifiers map the test data from the optimal attribute set of the trained dataset over the feature vector created by the classifiers during their training. Classifiers then group the data into DDoS and benign requests. We used Support Vector Machine (SVM), Random Forest (RF), K-nearest Neighbor (KNN), Multilayer Perceptron (MLP), AdaBoost (AB), and Decision Tree (DT) classifiers that were also discussed in base paper and two more best classifiers discussed in literature like Bayesan, J48 [19], [20].

4.5 Protocol Based Classification Results

Our proposed machine learning based defensive system is capable of detecting TCP, UDP and ICMP flooding attacks. NSL-KK data is used in our work. We selected minimum number of attributes as discussed in attribute selection module. To distinguish between ICMP, TCP and UDP flooding attacks, system is evaluated on the basis of protocol_type features (attributes) of NSL-KDD dataset. In protocol_type, three different types of traffic protocols are given, such as TCP, UDP and ICMP. There are two dynamic attributes are selected named as src_bytes and wrong_fragments. These attributes are maximum correlated to protocol_type. To handle TCP, UDP and ICMP flooding attacks in a cloud computation environment, it is important to distinguish them. We have classified the incoming traffic on the basis of class feature (attribute) and our proposed technique produced improved results in detection which are discussed in chapter 5.

4.6 Summary

In this chapter, proposed machine learning based classification of DDoS was discussed. NSL-KDD dataset is taken as input. This data is imported in Weka tool. There are 42 attributes present in the dataset. Data is normalized on a standard scale. 80% data is passed for training to select optimal attributes and 20% data is passed to the classifier. There are 9 attributes reported which are highly correlated. These 9 attributes are further classified through 8 different classifiers. The attacks are also classified on the basis of protocol to distinguish between tcp, udp and icmp taffic.

Chapter 5

Results and Discussion

In this chapter, experiment results of our proposed technique are discussed. Different performance calculating metrics are used for evaluation. These evaluation metrics are the same as discussed in base paper [18]. At the start, comparative analysis of all classifiers with multiple performance metrics. Comparative analysis of different classifiers on the basis of true attack detection rate, false attack detection rate, root mean square error (RMSE), precision, recall and F-score is discussed. Further, these results compared with base paper results. The research question [3] which is raised in chapter 1 is answered in 5.1.1.

5.1 Proposed Method Evaluation

Initially, the dataset is exported to Weka tool and normalizes to bring all attributes to a regular scale [0-1]. After preprocessing the data in the first module, 80% of the data is transferred to the second best attribute selection module for training and 20% for testing, to the third classification module. In the most correlated attribute selection module, using the two methods described in Chapter 4, the number of attributes decreased from 41 to 9 only. These important attributes are src_bytes, dst_bytes, logged_in, serror_rate, srv_rerror_rate, diff_srv_rate, srv_diff_host_rate,

[3] What are the parameters used by surveyed machine learning techniques to evaluate the performance of their technique?

dst_host_srv_diff_host_rate, dst_host_srv_error_rate and class. Further dataset is classified in with 8 different classifiers. Detailed discussion was presented in chapter 4.

5.1.1 Evaluation Metrics used for Computing Results

In order to measure the efficiency of the research, following parameters are used. These formulas are based on already published (base) paper [18]. Our experiment results are better and improved as compaerd to base paper results. In below listed equations, A, B, C, D, L and M. A presents correctly predicted normal class, B presents correctly predicted anomaly class, C presents incorrectly predicted normal class, D incorrectly predicted anomaly class. L and M are actual and predicted class respectively. These formulas are listed below.

- **Accuracy**: It is the percentage of correctly detection of normal and anomaly class in the given dataset. It is calculated using this equation 5.1.

$$Accuracy = \frac{(A+B)}{(A+B+C+D)} \tag{5.1}$$

- **True positive rate (TPR)**: It is the percentage of correctly detection of normal class in the given dataset. It is calculated using this equation 5.2.

$$TPR = \frac{A}{(A+D)} \tag{5.2}$$

- **True negative rate (TNR)**: It is the percentage of correctly detection of anomaly class in the given dataset. It is calculated using this equation 5.3.

$$TNR = \frac{B}{(B+C)} \tag{5.3}$$

- **False positive rate (FPR)**: It incorrectly indicate a normal class as anomaly class in the given dataset. It is calculated using this equation 5.4.

$$FPR = \frac{C}{(B + C)} \tag{5.4}$$

- **False negative rate (FNR)**: It incorrectly indicate an anomaly class as normal class in the given dataset. It is calculated using this equation equation 5.5.

$$FNR = \frac{D}{(A + D)} \tag{5.5}$$

- **Precision**: A technique can detect specific type of DDoS attacks than non-relevant ones. It is calculated by using equation 5.6.

$$Precision = \frac{TPR}{(TPR + FPR)} \tag{5.6}$$

- **Recall**: A technique can detect specific type of DDoS attacks than actual ones. It is calculated by using equation 5.7.

$$Recall = \frac{TPR}{(TPR + FNR)} \tag{5.7}$$

- **F-score**: It is a measure of a models accuracy on a dataset (NSL-KDD). It is the combining the precision and recall of the model, and it is defined as the harmonic mean of the models precision and recall. It can be calculated through this formula. It is calculated using this equation 5.8.

$$Fscore = 2 * \frac{(precision + recall)}{(precision * recall)} \tag{5.8}$$

- **RootMeanSquareError(RMSE)**: used to find error between actual and predicted class. It is calculated by using this equation 5.9.

$$RMSE = \sqrt{\frac{1}{n}\sum_{n=1}^{n}(L-M)^2} \tag{5.9}$$

There are eight different classifiers used in proposed technique. These six classifiers called Random Forest (RF), Multilayer Perceptrons (MLP), Support Vector Machine (SVM), K-Nearest Neighbor(KNN), Decision Tree (DT) and Adaboost (AD) are taken from base paper [18] and two additional efficient classifiers named J48 and Naive Bayes (NB) [56]. These are the different evaluation metrices used such as accuracy, true positive rate (TPR), false positive rate(FPR) , true negative rate(TNR), false negative rate(FNR), root mean square error(RMSE) and time taken(T.time(s)). The comparison of these seven performance metrices are used. Comparative analysis of these classifiers is shown in table 5.1.

TABLE 5.1: Comparative Analysis of Different Classifiers

Classifier	Accuracy%	TPR%	TNR%	FPR	FNR	RMSE	T.time(s)
MLP	94.4021	94.4	94.8	0.059	0.062	0.2228	1873.29
SVM	94.6017	94.6	94	0.059	0.0531	0.2323	121.26
RF	**98.7048**	**0.987**	**0.744**	**0.014**	**0.0145**	**0.0986**	**5.42**
AB	90.3744	90.4	93.4	0.093	0.067	0.2627	1.46
DT	97.3518	97.4	95.6	0.027	0.569	0.1382	17
KNN(ibk)	97.5825	97.6	97.6	0.026	0.025	0.1544	0.01
NB	80.731	80.7	94.8	0.158	0.294	0.4371	0.19
J48	**98.5983**	**0.986**	**0.989**	**0.014**	**0.018**	**0.1056**	**1.6**

To evaluate the detection rate of proposed research method, Accuracy, TPR%, TNR%, FPR and FNR parameters are taken to visualize the attack detection. Performance graph of these parameters is shown in figure 5.1. J48, RF, DT and KNN are the best classifier as they scored maximum and almost same. J48 and RF acheived up to 99% accuracy, attack detection rate up to 99% with very few false alert rate. J48 and RF achieve up to 99% accuracy, an attack identification

rate of up to 99% with very few false alarms. KNN and DT achieve up to 98% accuracy, an attack identification rate of up to 98% with very few false alarms. Therefore, J48 and RF are the two best classifiers which are producing maximum results.

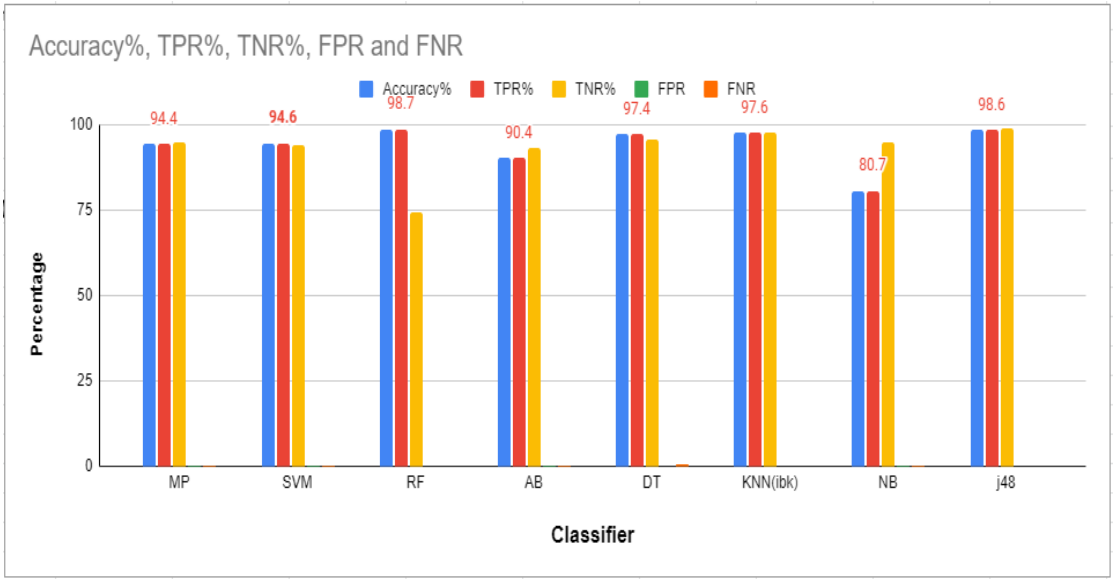

FIGURE 5.1: Performance of Different Classifiers

There is a comparison between multiple classifiers on the basis of correctly detection of attacks is shown in figure 5.2. This correct detection is divided into two parts; true positive detection (TPR) and true negative detection (TNR). Classifiers are plotted on the X-axis and rate of attack detection (%) is plotted on the y-axis. Green bar presents percentage of detection of truly occurring DDoS attacks (anomaly class) and Gray bar presents percentage (%) of detection of normal (class) traffic. On the top of the bar, percentage score of attack detection is mentioned. TPR of all classifiers except NB is greater than 90%. TNR of six classifiers such as MLP, SVM, AB, DT, KNN (ibk) and J48 is greater than 90 while RF has less than 75%. If we observe the overall performance of true detection rate of all classifiers, it is found that J48, KNN and DT are the three best classifiers which are producing higher than 95% TPR and TNR rate. RF is producing TPR of 98.7% which is good but, TNR is less than 75% which is a little bit low. So it is concluded that J48, DT and KNN are better classifiers than the other classifiers in term of accurately classification of normal and DDoS traffic.

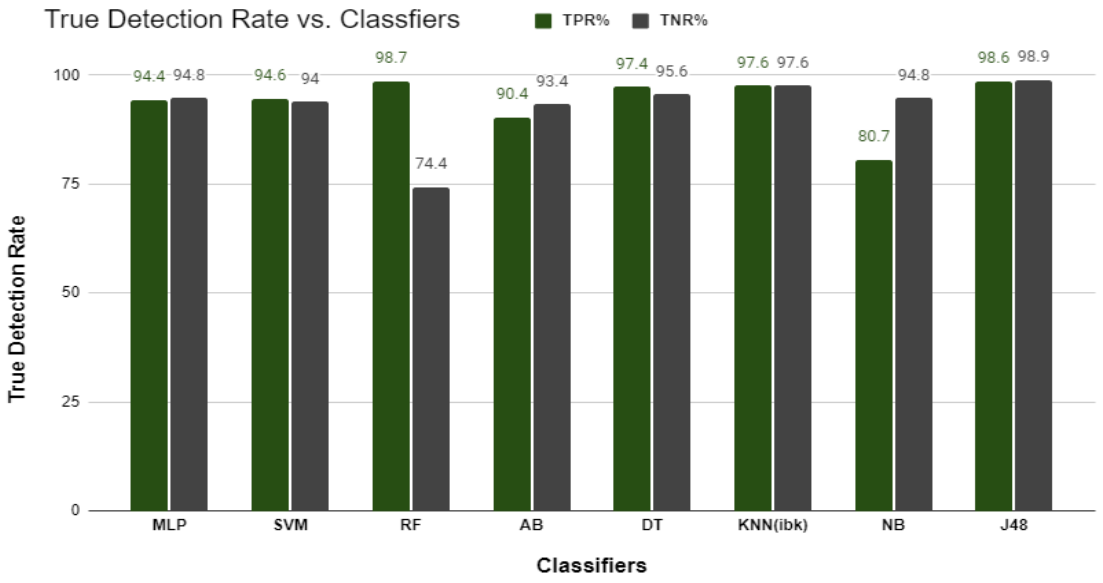

FIGURE 5.2: True detection rate of different classifiers

There is a comparison between different classifiers on the basis of false detection of attacks is shown in figure 5.3. This false detection is divided into two parts; false positive detection (FPR) and true negative detection (FNR). FP is the false prediction of DDoS as normal traffic. While FN is the false prediction of normal traffic as DDoS traffic. FPR is under 0.01 in all classifier other than NB. DT is generating maximum FNR of 0.569 while RF and J48 are generating minimum FNR of 0.014 and 0.018 respectively. The false attack detection rate is high in NB. RF, J48 and KNN (ibk) are producing minimum false attack detection.

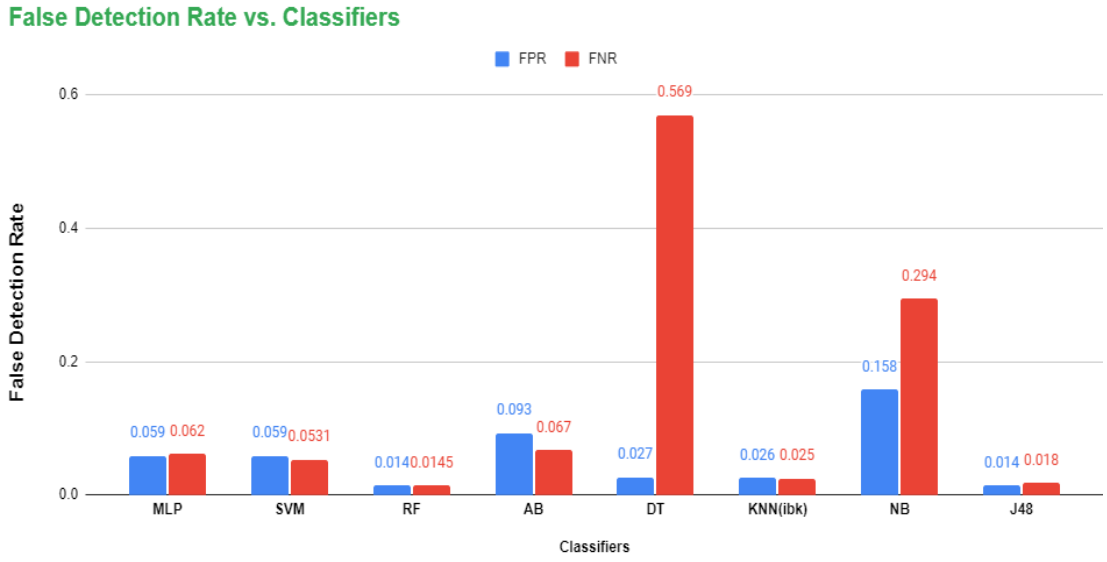

FIGURE 5.3: False Detection Rate of Different Classifiers

There is a comparison between different classifiers on the basis of the time taken in seconds by each classifier is shown in figure 5.4. Multilayer preceptrons (MLP) took 1873 seconds for classification and Support vector machine (SVM) took 121 seconds for classification. Red bar presents the highest time taken classifier MLP and Orange bar presents the 2nd highest time taken classifier SVM. These highest values are normalized on a standard scale [0-100]. It is important to bring these values in a normal scale so that time taken by other classifiers could be analyzed. KNN took minimum time of 0.01 seconds. AB, NB and j48 took less time to classify as compared to RF. Therefore, using MLP and SVM for the classification is not good idea.

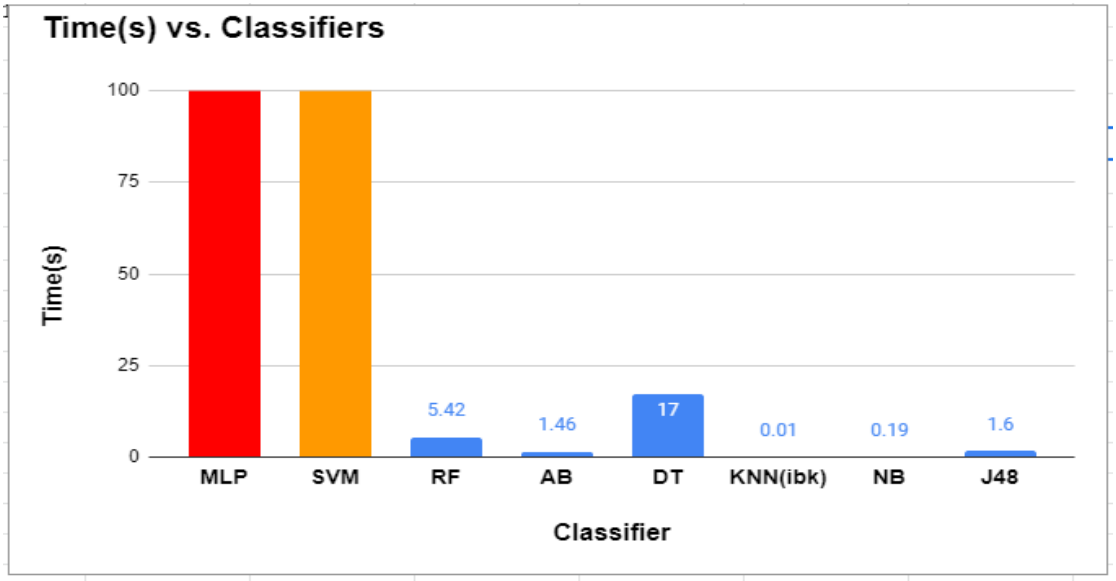

FIGURE 5.4: Comparative Analysis of T.time(s) of Different Classifiers

In order to check the attack prediction of our proposed system, we analyze Root mean square error (RMSE). It is used for calculating quantitative results is a common way to check a model's error. It is always non-negative and a value of 0 would indicate that it is perfectly suited for the data (might never in reality). A lower value is generally better than a higher value. In figure 5.5, 0 to 0.5 scale is set to analyze RMSE values of different tested classifiers. Naive Bayes (NB) classifier has a higher value of 0.44 among all other classifiers and Random Forest (RF) and J48 has a minimum value of 0.1. Therefore, it is concluded that the rate of error is high in NB as compared to RF and J48.

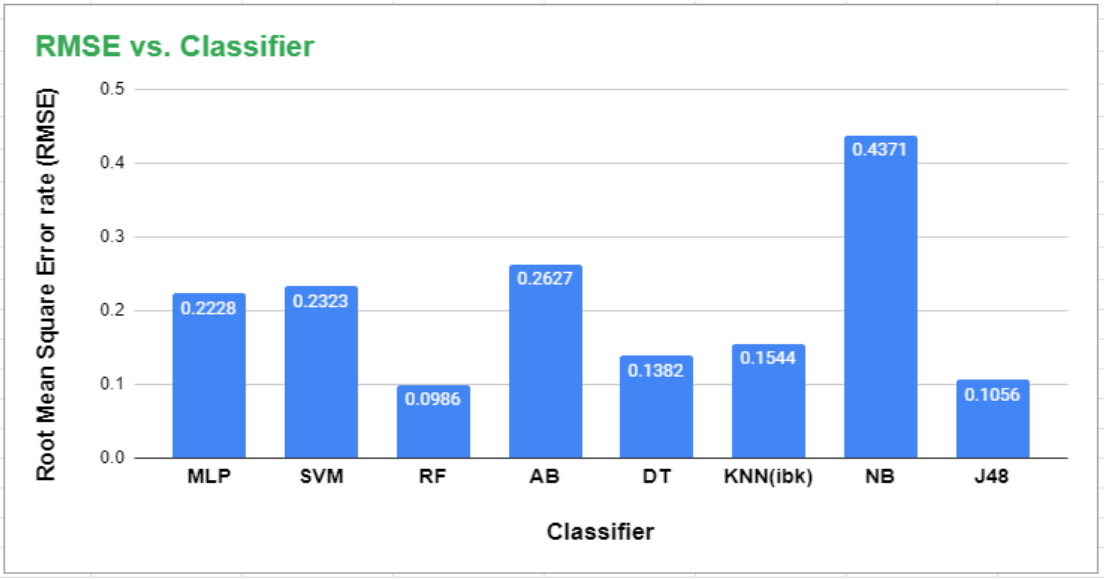

FIGURE 5.5: Root Mean Square Error (RMSE)

5.1.2 Protocol Based Classification Results

To handle TCP, UDP and ISMP flooding attacks in cloud computation environment, it is important to distinguish them. We have classified the incoming traffic on the basis of class feature (attribute). TCP, UDP and ICMP are the protocol types. J48 and RF (which are the two best selected classifiers under multiple scenarios) are used. These two classifiers are able to classify TCP and UDP protocols with 99% true detection, precision, recall and F-score and have very low false detection rate. But in ICMP, evaluation scores are slightly low. For ICMP protocol, the true detection rate of J48 classifier is 95.6% while RF has a detection rate of 96.3% which is a little bit improved. Precision is also a litte bit improved in RF than J48. All other evaluation parameters are almost same for these two classifiers J48 and RF are listed in table 5.2.

TABLE 5.2: Protocol based Classification of Flooding Attacks

Classifiers	Protocols	TP rate	FP rate	Precision	Recall	F-score
J48	**TCP**	99.5	0.011	99.8	99.5	99.6
	UDP	99	0.006	95.9	99	97.4
	ICMP	95.8	0.001	98.6	95.8	97.3
RF	**TCP**	99.5	0.010	99.8	99.5	99.7
	UDP	99	0.006	95.9	99	97.4
	ICMP	96.3	0.001	98.6	96.3	97.4

5.2 Comparison of Results with Published (base) Paper

In this section, the experiment results of our proposed approach are compared with the best technique given in base paper. Results of different approaches discussed in base paper are shown in table 5.3. When we compare our results with the results that were reported in base paper, found that our proposed system is slightly better in performance. The experimental results of our proposed techniques are shown in 5.1. In base paper [18], most effective results are produced by selecting mean absolute deviation method (MAD) along with Random Forest (RF) classifier. But we use CFsSubsetEval (using BestFirstSearch method) along with Random Forest (RF) and J48 produced a much better results. It is concluded from our results that J48 and Random Forest (RF) is the better DDoS classifiers to classify anomaly and normal class with high accuracy, precision, recall and F-score. J48 discovered as a great competitor of Random Forest (RF) and it takes less time to classify.

TABLE 5.3: Results of Different Classifiers with MAD [18] Technique

Classifier	Accuracy%	TPR%	TNR%	FPR	FNR	RMSE	T.time(s)
SVM	91.44	95.03	85.36	0.1463	0.0496	0.2925	124.504
RF	**98.226**	**98.32**	**98.066**	**0.019**	**0.0167**	**0.1398**	**0.739**
KNN	96.31	98.52	92.56	0.0743	0.0147	0.192	12.355
AB	95.76	97.56	92.71	0.0728	0.0243	0.2057	6.404
MLP	97.07	98.58	94.51	0.0548	0.0141	0.171	203.961
DT	96.19	98.03	93.08	0.0691	0.0196	0.167	0.603

In table 5.4, protocol based classification is done using Mean absolute deviation (MAD) with Random Forest (RF) classifier that is discussed in base paper [18]. The reason for reporting it here to compare our protocol based classification results that were listed in table 5.2. It is observed that there is a significant improvement in protocol based classification using Random Forest (RF) classifier. True detection rate of TCP protocol is 99.5% while in base paper, it is 98.5%. True detection rate of UDP protocol is 99 while in base paper, it is 93.45%. True detection rate of ICMP protocol is 96.3% while in base paper, it is 95.8%. Results are improved in our suggested approach as listed in 5.2. Results of J48 classifier also competing the results of Random Forest (RF) in all aspects. So, it is concluded that J48 and

RF are the best classifiers to detect protocol based TCP, UDP and ICMP flooding attacks.

TABLE 5.4: Protocol based Classification using MAD-RF [18]

Protocols	**Accuracy%**	**TPR%**	**TNR%**	**FPR%**	**FNR%**
UDP	93.452	93.746	50	0.5	0.062
ICMP	95.869	66.66	99.716	0.0028	0.33
TCP	98.585	99.451	97.232	0.027	0.0054

Correlation feature selection (CFS) machine learning technique along with Random Forest (RF) and J48 are producing better results as compared to other 6 classifiers. Using CFS for attribute selection reduced the number of testing attributes to 9 (most correlated) which improves results of Random Forest (RF) classifier as listed in base paper [18]. Further, Random Forest (RF) and J48 produced a significant improvement in the detection rate of protocol based traffic like TCP, UDP and ICMP as compared to results in base paper [18].

Chapter 6

Conclusions and Future Work

Our proposed machine learning based classification technique produced improved results to detect DDoS attack. Our proposed technique is based on three modules; pre-processing, attributes selection and detection and prevention system. At the start, all attributes of the incoming traffic are normalized on a standard scale to apply different machine learning techniques. CFsSubsetEval along with BestSearchFirst output most correlated features. The number of features is reduced from 41 to 9 and these selected features are most correlated to class feature (normal/anomaly). Random Forest (RF) and J48 achieved maximum results in term of accuracy (98.7%), true attack detection rate (98.5%), minimum false detection rate (0.015). False attack detection rate and root mean square rate is minimum in these two classifiers. But J48 took less time than RF. K-nearest Neighnor (KNN) and Decision Tree (DT) have also achieved maximum accuracy of 97.5% with low false detection rate.

It is concluded that our proposed machine learning DDoS classification produced high rate of detection of 98.7%. Random Forest (RF) produced 98.7% of accuracy which is better than results discussed in base paper [18]. J48 is other one classifier which is a good competitor of Random Forest (RF) and it takes less time to classify as compared to RF. KNN(ibk) is the 3rd good classifier and DT is the 4th good classifier found which are producing close results in all aspects as compared to RF and J48 classifiers.

6.1 Future Work

Our proposed machine learning based classification technique is detecting DDoS attack with high detection rate and low false detection rate. But, this technique is tested on NSL-KDD dataset. This technique could be tested on other datasets. Correlation feature selection (CFS) machine learning technique is used to select minimum and most correlated attributes in our proposed technique. Other attributes selection techniques could be added. Other classifiers could also be added to improve attack detection with low false attack detection.

Bibliography

[1] A. Alsirhani, S. Sampalli, and P. Bodorik, "Ddos attack detection system: Utilizing classification algorithms with apache spark," in *2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. IEEE, 2018, pp. 1–7.

[2] O. Osanaiye, H. Cai, K.-K. R. Choo, A. Dehghantanha, Z. Xu, and M. Dlodlo, "Ensemble-based multi-filter feature selection method for ddos detection in cloud computing," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, pp. 1–10, 2016.

[3] O. A. Wahab, J. Bentahar, H. Otrok, and A. Mourad, "Optimal load distribution for the detection of vm-based ddos attacks in the cloud," *IEEE transactions on services computing*, vol. 13, no. 1, pp. 114–129, 2017.

[4] R. V. Deshmukh and K. K. Devadkar, "Understanding ddos attack & its effect in cloud environment," *Procedia Computer Science*, vol. 49, pp. 202–210, 2015.

[5] N. Agrawal and S. Tapaswi, "Defense mechanisms against ddos attacks in a cloud computing environment: State-of-the-art and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3769–3795, 2019.

[6] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and R. Buyya, "Service resizing for quick ddos mitigation in cloud computing environment," *Annals of Telecommunications*, vol. 72, no. 5, pp. 237–252, 2017.

[7] N. Agrawal and S. Tapaswi, "Defense schemes for variants of distributed denial-of-service (ddos) attacks in cloud computing: A survey," *Information Security Journal: A Global Perspective*, vol. 26, no. 2, pp. 61–73, 2017.

[8] G. Huston, R. Hinden, O. Troan, F. Gont, and I. F. C. Fragile, "Internet area wg r. bonica internet-draft juniper networks intended status: Best current practice f. baker expires: January 24, 2019 unaffiliated," 2018.

[9] S. Gnanambal, M. Thangaraj, V. Meenatchi, and V. Gayathri, "Classification algorithms with attribute selection: an evaluation study using weka," *International Journal of Advanced Networking and Applications*, vol. 9, no. 6, pp. 3640–3644, 2018.

[10] C. Yang, "Anomaly network traffic detection algorithm based on information entropy measurement under the cloud computing environment," *Cluster Computing*, vol. 22, no. 4, pp. 8309–8317, 2019.

[11] Z. He, T. Zhang, and R. B. Lee, "Machine learning based ddos attack detection from source side in cloud," in *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*. IEEE, 2017, pp. 114–120.

[12] R. Singh, H. Kumar, and R. Singla, "An intrusion detection system using network traffic profiling and online sequential extreme learning machine," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8609–8624, 2015.

[13] S.-C. Tsai, I.-H. Liu, C.-T. Lu, C.-H. Chang, and J.-S. Li, "Defending cloud computing environment against the challenge of ddos attacks based on software defined network," in *Advances in Intelligent Information Hiding and Multimedia Signal Processing*. Springer, 2017, pp. 285–292.

[14] A. R. Wani, Q. Rana, U. Saxena, and N. Pandey, "Analysis and detection of ddos attacks on cloud computing environment using machine learning techniques," in *2019 Amity International conference on artificial intelligence (AICAI)*. IEEE, 2019, pp. 870–875.

[15] K. Bhushan and B. B. Gupta, "Distributed denial of service (ddos) attack mitigation in software defined network (sdn)-based cloud computing environment," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 5, pp. 1985–1997, 2019.

[16] S. Dong, K. Abbas, and R. Jain, "A survey on distributed denial of service (ddos) attacks in sdn and cloud computing environments," *IEEE Access*, vol. 7, pp. 80 813–80 828, 2019.

[17] Z. Chen, F. Jiang, Y. Cheng, X. Gu, W. Liu, and J. Peng, "Xgboost classifier for ddos attack detection and analysis in sdn-based cloud," in *2018 IEEE international conference on big data and smart computing (bigcomp)*. IEEE, 2018, pp. 251–256.

[18] P. Verma, S. Tapaswi, and W. W. Godfrey, "An adaptive threshold-based attribute selection to classify requests under ddos attack in cloud-based systems," *Arabian Journal for Science and Engineering*, vol. 45, no. 4, pp. 2813–2834, 2020.

[19] V. Mhetre and M. Nagar, "Classification based data mining algorithms to predict slow, average and fast learners in educational system using weka," in *2017 International Conference on Computing Methodologies and Communication (ICCMC)*. IEEE, 2017, pp. 475–479.

[20] S. A. Kiranmai and A. J. Laxmi, "Data mining for classification of power quality problems using weka and the effect of attributes on classification accuracy," *Protection and Control of Modern Power Systems*, vol. 3, no. 1, pp. 1–12, 2018.

[21] B. Gupta and O. P. Badve, "Taxonomy of dos and ddos attacks and desirable defense mechanism in a cloud computing environment," *Neural Computing and Applications*, vol. 28, no. 12, pp. 3655–3682, 2017.

[22] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and R. Buyya, "Service resizing for quick ddos mitigation in cloud computing environment," *Annals of Telecommunications*, vol. 72, no. 5, pp. 237–252, 2017.

[23] V. C. Pandey, S. K. Peddoju, and P. S. Deshpande, "A statistical and distributed packet filter against ddos attacks in cloud environment," *Sādhanā*, vol. 43, no. 3, pp. 1–9, 2018.

[24] S. D. C. Ramakrishna, "Prevention of ddos & edos using hybrid filtering technique in a cloud environment shruti wadhwa1 and dr. poonam," *International Journal of Pure and Applied Mathematics*, vol. 114, no. 12, pp. 383–392, 2017.

[25] A. Bremler-Barr, E. Brosh, and M. Sides, "Ddos attack on cloud auto-scaling mechanisms," in *IEEE International Conference on Computer Communications 2017-IEEE Conference on Computer Communications*. IEEE, 2017, pp. 1–9.

[26] A. Aborujilah and S. Musa, "Cloud-based ddos http attack detection using covariance matrix approach," *Journal of Computer Networks and Communications*, vol. 2017, 2017.

[27] B. Gupta and O. P. Badve, "Garch and ann-based ddos detection and filtering in cloud computing environment," *International Journal of Embedded Systems*, vol. 9, no. 5, pp. 391–400, 2017.

[28] M. Idhammad, K. Afdel, and M. Belouch, "Detection system of http ddos attacks in a cloud environment based on information theoretic entropy and random forest," *Security and Communication Networks*, vol. 2018, 2018.

[29] A. Bhardwaj, G. Subrahmanyam, V. Avasthi, and H. G. Sastry, "Solutions for ddos attacks on cloud," in *2016 6th International Conference-Cloud System and Big Data Engineering (Confluence)*. IEEE, 2016, pp. 163–167.

[30] K. Bhushan and B. Gupta, "Hypothesis test for low-rate ddos attack detection in cloud computing environment," *Procedia computer science*, vol. 132, pp. 947–955, 2018.

[31] Z. Li, H. Jin, D. Zou, and B. Yuan, "Exploring new opportunities to defeat low-rate ddos attack in container-based cloud environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 31, no. 3, pp. 695–706, 2019.

[32] J. Jiao, B. Ye, Y. Zhao, R. J. Stones, G. Wang, X. Liu, S. Wang, and G. Xie, "Detecting tcp-based ddos attacks in baidu cloud computing data centers," in *2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS)*. IEEE, 2017, pp. 256–258.

[33] A. Rukavitsyn, K. Borisenko, and A. Shorov, "Self-learning method for ddos detection model in cloud computing," in *2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. IEEE, 2017, pp. 544–547.

[34] K. Borisenko, A. Smirnov, E. Novikova, and A. Shorov, "Ddos attacks detection in cloud computing using data mining techniques," in *Industrial Conference on Data Mining*. Springer, 2016, pp. 197–211.

[35] G. A. Sophia and M. Gandhi, "Stealthy ddos detecting mechanism for cloud resilience system," in *2017 International Conference on Information Communication and Embedded Systems (ICICES)*. IEEE, 2017, pp. 1–5.

[36] S. Borah, R. Panigrahi, and A. Chakraborty, "An enhanced intrusion detection system based on clustering," in *Progress in Advanced Computing and Intelligent Engineering*. Springer, 2018, pp. 37–45.

[37] E. De la Hoz, E. De La Hoz, A. Ortiz, J. Ortega, and B. Prieto, "Pca filtering and probabilistic som for network intrusion detection," *Neurocomputing*, vol. 164, pp. 71–81, 2015.

[38] İ. Özçelik and R. R. Brooks, "Deceiving entropy based dos detection," *Computers & Security*, vol. 48, pp. 234–245, 2015.

[39] P. Ghosh, A. Karmakar, J. Sharma, and S. Phadikar, "Cs-pso based intrusion detection system in cloud environment," in *Emerging Technologies in Data Mining and Information Security*. Springer, 2019, pp. 261–269.

[40] İ. Özçelik and R. R. Brooks, "Cusum-entropy: an efficient method for ddos attack detection," in *2016 4th International Istanbul Smart Grid Congress and Fair (ICSG)*. Ieee, 2016, pp. 1–5.

[41] M. H. Bhuyan, D. Bhattacharyya, and J. K. Kalita, "An empirical evaluation of information metrics for low-rate and high-rate ddos attack detection," *Pattern Recognition Letters*, vol. 51, pp. 1–7, 2015.

[42] J. Cao, B. Yu, F. Dong, X. Zhu, and S. Xu, "Entropy-based denial-of-service attack detection in cloud data center," *Concurrency and Computation: Practice and Experience*, vol. 27, no. 18, pp. 5623–5639, 2015.

[43] C. Guo, Y. Ping, N. Liu, and S.-S. Luo, "A two-level hybrid approach for intrusion detection," *Neurocomputing*, vol. 214, pp. 391–400, 2016.

[44] Z. Jian-Qi, F. Feng, Y. Ke-Xin, and L. Yan-Heng, "Dynamic entropy based dos attack detection method," *Computers & Electrical Engineering*, vol. 39, no. 7, pp. 2243–2251, 2013.

[45] K. Lee, J. Kim, K. H. Kwon, Y. Han, and S. Kim, "Ddos attack detection method using cluster analysis," *Expert systems with applications*, vol. 34, no. 3, pp. 1659–1665, 2008.

[46] B. Wang, Y. Zheng, W. Lou, and Y. T. Hou, "Ddos attack protection in the era of cloud computing and software-defined networking," *Computer Networks*, vol. 81, pp. 308–319, 2015.

[47] M. Zareapoor, P. Shamsolmoali, and M. A. Alam, "Advance ddos detection and mitigation technique for securing cloud," *International Journal of Computational Science and Engineering*, vol. 16, no. 3, pp. 303–310, 2018.

[48] B. K. Devi and T. Subbulakshmi, "Cloud-based ddos attack detection and defence system using statistical approach," *International Journal of Information and Computer Security*, vol. 11, no. 4-5, pp. 447–475, 2019.

[49] R. Kesavamoorthy and K. R. Soundar, "Swarm intelligence based autonomous ddos attack detection and defense using multi agent system," *Cluster Computing*, vol. 22, no. 4, pp. 9469–9476, 2019.

[50] M. H. Bhuyan, D. Bhattacharyya, and J. K. Kalita, "E-ldat: a lightweight system for ddos flooding attack detection and ip traceback using extended

entropy metric," *Security and Communication Networks*, vol. 9, no. 16, pp. 3251–3270, 2016.

[51] A. N. Jaber, M. F. Zolkipli, H. A. Shakir, and M. R. Jassim, "Host based intrusion detection and prevention model against ddos attack in cloud computing," in *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*. Springer, 2017, pp. 241–252.

[52] H. Pillutla and A. Arjunan, "Fuzzy self organizing maps-based ddos mitigation mechanism for software defined networking in cloud computing," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 4, pp. 1547–1559, 2019.

[53] R. Saxena and S. Dey, "Ddos attack prevention using collaborative approach for cloud computing," *Cluster Computing*, pp. 1–16, 2019.

[54] B. Paharia and K. Bhushan, "Ddos detection and mitigation in cloud via fogfiter: a defence mechanism," in *2018 9th international conference on computing, communication and networking technologies (ICCCNT)*. IEEE, 2018, pp. 1–7.

[55] S. Gnanambal, M. Thangaraj, V. Meenatchi, and V. Gayathri, "Classification algorithms with attribute selection: an evaluation study using weka," *International Journal of Advanced Networking and Applications*, vol. 9, no. 6, pp. 3640–3644, 2018.

[56] A. R. Onik, N. F. Haq, L. Alam, and T. I. Mamun, "An analytical comparison on filter feature extraction method in data mining using j48 classifier," *International Journal of Computer Applications*, vol. 124, no. 13, 2015.

[57] M. Doshi, "Correlation based feature selection (cfs) technique to predict student perfromance," *International Journal of Computer Networks & Communications*, vol. 6, no. 3, p. 197, 2014.

[58] L. Dhanabal and S. Shantharajah, "A study on nsl-kdd dataset for intrusion detection system based on classification algorithms," *International Journal*

of Advanced Research in Computer and Communication Engineering, vol. 4, no. 6, pp. 446–452, 2015.