

A Fuzzy Logic based Defense Mechanism against Distributed Denial of Service Attack in Cloud Computing Environment

N.Ch.S.N. Iyengar¹, Arindam Banerjee² and Gopinath Ganapathy³

^{1,2}School of Computing Science and Engineering, VIT University, Vellore-632014, Tamil Nadu, India.

³Technology Park Bharathidasan University, Trichy-620023, India

nchsnnyengar48@gmail.com, arindam.banerjee2012@vit.ac.in, gganapathy@gmail.com

Abstract: Cloud defines a new age of computing solution that provides services to customers with its unique features of agility and multi-tenancy. As the critical resources are hosted at cloud provider's end, security is a big challenge in cloud computing. If the cloud environment is compromised and attackers get the access of core data centers, the availability of the critical resources becomes a big concern for the service consumers. Denial of Service and Distributed Denial of Service kind of attacks are launched towards cloud environment to make the resources unavailable for legitimate users. In this paper we propose a fuzzy logic based defense mechanism that can be set with predefined rules by which it can detect the malicious packets and takes proper counter measures to mitigate the DDoS attack. Also a detailed study of different kind of DDoS attack and existing defense strategies has been carried out.

Keywords: DoS, DDoS, fuzzy logic, anomaly detection, entropy, Http packet.

1. Introduction

Cloud computing is an emerging field that presents new consumption and delivery model for virtualized IT solutions. Cloud encapsulates several existing web technologies to offer reduced management, improved scalability and on demand availability of resources. Agile deployment model, low investment, multi tenancy are all added attributes of cloud technology. In cloud environment, users' data and applications are scattered at remote data centers that can be accessed at users' end in virtualized form irrespective of users' locations and time. But the advantages of on demand resource availability in cloud environment come with several security issues. As users' confidential data are stored and applications are deployed at cloud service provider's end, continual service availability and protection of sensitive data are big concerns to consider for consumers.

The ubiquitous usage of cloud based service has brought remotely deployed applications in all kind of client devices using web based technologies. In the same time it is also susceptible to the same attacks and reliability problems that plague other IP-based data and web services. Unavailability of services and connectivity issues [22] in cloud can disrupt the service completely which tolls huge business loss for consumers. Denial of service (DoS) and distributed denial of service (DDoS) are two web based attacks aiming to make critical resource unavailable to legitimate users.

DDoS, which is an amplified and advanced form of DoS, is the security breach that targets the remote data centers running important services and floods the servers with huge amount of packets that is unbearable to the victim server

causing unavailability of services to legitimate users. DDoS is a tempting way to attack the service providers due to the wide spread availability of attack tool and simplicity of the attack strategy. The presence of DDoS attack in web history can be traced back to its first occurrence in June, 1998 [5]. However denial of service attacks such as ICMP/Ping flood that stops legitimate users from accessing network resource was known to network research community in 1980s. In February 2000, a Canadian hacker namely "mafiaboy" [1] successfully launched a series of denial of service attacks against several commercial sites, including Yahoo, Amazon, Fifa and eBay, which is considered as first documented DoS attack in history causing an estimated cumulative loss of US\$1.2 billion. Due to the DDoS attack on its site in January, 2001, Microsoft faced a disruption of service in its web sites including Microsoft.com, CarPoint.com, MSN.com, Encarta.com and Expedia.com [25]. In October 2002, an hour long, sophisticated DDoS attack crippled nine of the thirteen geographically spread servers that used to manage Domain Name System (DNS) service to users [26]. In January 2004, the Mydoom worm was used to carry out DDoS attack against SCO group following which SCO group announced a bounty of US\$250,000 "for information leading to the arrest and conviction of the individual or individuals responsible for creating the Mydoom virus" [27]. It was estimated that globally around one million computers were infected by massive spread of Mydoom worm. Denial of service has gained its popularity with distributed nature to conduct against commercial sites. In recent times, March 2013, Spamhaus Project was affected by a massive 300 Gbps packet flood attack which was recovered by CloudFlare [28]. In February 2014, CloudFlare reported that it mitigated the worst DDoS attack ever against a French site that reached upto 400Gbps of packet flood using NTP amplification [29]. The web traffic analysis reveals that, on average 1.29 DDoS attacks are occurring worldwide in every two minutes [24]. As the data are distributed in cloud environment, DDoS attack in cloud is on the rise in every year with available network attack tools. Trin00 or Trinoo [23] is considered to be first tool or set of programs written in C that was used extensively by hackers to launch several DDoS attacks. DDoS attack softwares or tools simulate a huge number of packet requests concurrently to victim server preventing legitimate customers from visiting the site. Now a days, DDoS attacks are initiated by the help of widely scattered, networked botnets or zombies that simultaneously send a

huge amount of traffic to target machine. If this attack persists for long time, it also prevents search engine spiders from visiting the website that causes loss of its page ranking so that potential customers would no longer be able to find the website by using major search engines. In general, DoS activity can be seen as an act of vandalism that instigates a cyber-attack having experimental, financial, technological, political or socio-economical motive.

Availability of multiple Denial of Service attack tools attracts inexperienced hackers commonly referred to as Script Kiddies with limited technical skill. But the advancement of the DDoS defense mechanisms, mitigation tactics and strong security policies designed by experts left no space for novice attackers to gain much. Experienced and professional hackers' communities always research on existing security breach or flaws in protocol to design more complicated and advanced attack strategy. Different defense mechanisms against DDoS attacks have been discussed and proposed in literature. Also defense strategies have been classified based upon the classification DDoS attacks. Leland et al. [6] demonstrated network traffic models based on self-similar stochastic processes. This self-similarity is very essential in detecting denial of service scenario in a network. The objective of this paper is to discuss different kinds of existing DDoS attacks and related mitigation techniques. We also plan to design a fuzzy logic based model as a DDoS protection mechanism.

2. Motivation Behind DDoS Attack

Risk quantification and analysis provide several methodologies for categorizing and approximating security risk factors, estimating possible defense mechanism and their effectiveness to reduce risks. Designing an effective defense strategy needs a thorough study of attackers' inspiration behind launching a DDoS attack [3]. The attack mitigation method should be developed by considering plausible attack methods, attack opportunities and attack motives. DDoS attack can commonly be divided into below mentioned categories:

Experimental: This is the most common incentive for attackers to launch DDoS attack which is often carried out by rookie hackers with some easily available tools to get experience or just for fun. However this kind of attack often can be detected at the beginning stage itself and the traffic can be easily separated from the victim server. An efficient defense mechanism and strong cyber laws are useful for discouraging this kind of practice.

Competition: This is again a common reason for launching DDoS attacks against commercial firms for gaining financial and economical [9] profits in market by rival parties. But in this case the professionally and technically experienced hackers are hired and the attack becomes really dangerous, persistent and challenging to mitigate. These persistent attacks disrupt the vital services, damage the regular sales resulting into harm to reputation.

Revenge: DDoS attacks for taking revenge are often carried out by unhappy customers, frustrated employees or disgruntled hackers. Later the attacker might demand payment to stop the DoS attack for making profit.

Political reasons: DDoS attacks are also a weapon to

interrupt services and cyber infrastructure due to political incentives and ideologies [7]. DDoS attacks on US military sites in 2001 by Chinese hackers' group "Honker Union", series of DoS attack on Estonia government's web servers in 2007 [2], attack on CNN's site by a number of attack tools in 2008, DDoS attack on Burma's site Democratic Voice of Burma (DVB) in 2008 etc. are some of the incidents of DDoS attack due to political reasons [8].

Cyber war: This is another major incentive to launch powerful DDoS attack by hacktivists, military or terrorist organizations with a goal to halt the daily business and essential services of other countries. Highly experienced and skilled hackers are involved in this kind of attack to paralyze a country's daily online activities and vital services causing huge socio-economical loss. The attackers generally target privately and publicly available services, banking and finance organizations, state owned services and web servers, state-run organizations, telecommunication and mobile service vendors, energy and transportation infrastructure, healthcare corporations etc.

Apart from these major motivations some other common incentives can be – hackers, keen to establish reputations in cyber world, tests conducted by government or private security agencies, random attacks generated by flash crowd, self-induced accidental attack etc.

3. DDoS Attack Generation Tools

There are a number of DDoS attack tools developed for different operating platform that can be easily downloaded from web and can be used to launch a DDoS attack immediately. This attracts new hackers to play with DDoS attack against commercial firms. However attacks launched without much strategy and technical skill from attacker's side are often found as harmless or can easily be traced back to take any legal action against the attacker. Here we would discuss some widely available DDoS tools to launch attacks:

Trinoo or Trin00 [30]: It was the first well known tool used for launching DDoS attack by packet flooding from multiple machines. Trinoo was probably set up on thousands of machines connected on Internet that and compromised by "remote buffer overrun exploitation" [31]. Trinoo network comprises of attackers, master server, daemon and victim server. Master servers used to come under direct control of attackers and each master used to control a number of daemons. Later daemons were reasonable to conduct a consolidated packet flooding attack towards victim server.

Wintrinoo [30]: This is the windows version of Trinoo that increased the opportunity of more effective attack by compromising widely available machines run on windows operating system. It used to reach to user's end via email attachment and could be run by document macros.

LOIC (Low Orbit Ion Canon): LOIC is an open source software [32] available on internet and one of the most popular tool used for TCP, UDP packet flooding attack and testing network load. First written in C#, LOIC also has a version developed in JavaScript and a web browser version [33]. LOIC does not conceal attacker's IP address and hence it can be easily traced back.

HOIC (High Orbit Ion Canon): Like LOIC, it is also an open source network stress testing tool which can launch DDoS attack by means of HTTP flood also.

R-U-Dead-Yet (RUDY): RUDY is an open source [34] tool used to initiate a DoS attack by slow rate HTTP POST requests. The attack is accomplished by “long form field submission” [35] that injects one byte of information into POST request and then web application waits for these never ending POSTs. Attacker exhausts the server’s resources by creating a number of concurrent application threads.

XOIC: It is an openly available tool with a simple GUI, used to make DDoS attack to any IP address with user selected port and protocols (TCP, UDP, HTTP, ICMP). XOIC is considered to be more powerful than LOIC in some cases.

HULK (HTTP Unbearable Load King): It is a web server DDoS tool used to generate a chunk of unique and obfuscated traffic to strike server’s core resources. With each and every request, it generates some unique pattern to bypass server’s anomaly detection mechanism that looks for statistics on network traffic.

Tor’s Hammer: Similar to RUDY, it is also used to launch slow rate HTTP POST request attack [36]. Tor’s Hammer [37] is written in python which supports initiating attacks from random source IP address making it difficult to trace back the source machine of the attack.

Some other openly available important DDoS attack tools are: DDOSIM—Layer 7 DDOS Simulator, PyLoris, OWASP DOS HTTP POST, DAVOSET, GoldenEye HTTP Denial of Service Tool etc. A thorough understanding of currently available attack tools and their functionalities is necessary for designing an effective defense strategy.

4. Classification of DDoS Attack and Defense Mechanism

In the current scenario, cloud computing is the abstraction of services and securing a cloud service includes securing it from virtual machine vulnerabilities and service integration flaws. Attackers try to consume bandwidth, processing power and storage systems. DDoS is not a particular type of network attack but a common terminology to represent a group of attacks. As attackers target the loopholes of existing protocols at different network layers, the DDoS attack scenarios are classified based upon network layers.

In general DDoS attacks are categorized into two categories: i) layer-3 attack or network-transport level DDoS attack and ii) layer-7 attack or application level DDoS attack. We will discuss about these categories and their sub categories in detail.

4.1 Layer-3 attack / network-transport level DDoS attack

Layer-3 attacks are generally carried out to exhaust server’s resources by deploying high volume (number) of packets of TCP, UDP, ICMP protocols.

Flood Attack: Attackers bombard a large volume of packets to saturate the server’s network resources and eventually bring down the cloud service to a halt. UDP flood, ICMP flood, DNS flood are the widely used DDoS flood in layer-3 attack [17]. UDP flood attack leverages UDP packets to congest random or specific ports of the server keeping the server application busy in listening at the ports and when it does not find any application waiting for that ports it ultimately sends a “destination unreachable” ICMP message to spoofed source addresses. In ICMP flood attack, a large

burst of ICMP echo packet (‘Ping’ flood) is sent to destination that congest the bandwidth of server’s bandwidth as victim needs to reply all echo requests. Based upon the severity of the attack, the server side services can be slow or completely crashed down. UDP and ICMP floods are detectable and can be prevented by setting threshold values at border routers where routers only allow UDP/ICMP packets up to threshold rate. Daan van der Sanden et al. [18] proposed a mechanism to detect UDP attacks based on packet symmetry in UDP traffic flow.

TCP flood is another kind of DDoS flood attack at this level that makes victim server unable to respond to legitimate requests for new TCP connections [41]. There are several variations of TCP flood attack but TCP SYN flood is mostly used in DDoS attack that exploits the basis of 3-way handshake of TCP connection. Attackers use spoofed address to send several SYN packets containing TCP request to victim and server allocates Transmission Control Blocks. The server kernel memory gets exhausted and once the limit of half open connection is achieved server discards all other connection setup requests from legitimate users [42].

Reflection Attack: In reflection based DDoS attack, attacker sends requests to target victim as well as other machines in the network. The request packet carries the spoofed address of the victim and hence all other machines reply back to victim’s address making its bandwidth exhausted [19].

Amplification Attack: In this kind of attack, message volume is multiplied for each message and traffic towards victim is exaggerated. Amplification attack like SMURF attack sends ICMP ping requests to a network’s broadcast router in order to relay the message to all machines connected to that network [39]. Attacker spoofs the source IP address of the ICMP message to be the victim’s server so that all the responses go back to victim’s device. More the devices are there behind the border router more the attack is multiplied and becomes devastating in nature.

There is another variation of SMURF attack which is known as Fraggle Attack. In contrast to SMURF attack that sends ICMP Echo messages, fraggle attack sends UDP Echo messages to the ports supporting character generation.

DNS amplification is also a widely used DDoS attack strategy where attackers exploit the feature of DNS response being “substantially larger” than DNS request message [40]. DNS request is sent to an open DNS server making the source of the request spoofed to be the target’s IP address and large volume of response traffic is diverted to victim server.

4.2 Layer-7 attack / application level DDoS attacks

The application level or layer 7 DDoS attack has become a trend in now a days and its versatile nature has made it tough to be detected by anti-DDoS filters. In contrast to layer-3 attack, application level DDoS attack is more sophisticated and generally consumes less bandwidth with requests similar to legitimate ones. These attacks exploits the vulnerabilities of application level protocols and exhaust victim server’s computing resources by well-known applications such as domain name system (DNS), IRC, http, VoIP, SIP etc. Some of the widely experienced layer-7 DDoS attacks are given below:

Http flooding attack: Attackers mimic http requests of the legitimate users and overwhelm the server's resources by their request messages [20] so that the offered service by the victim server is delayed or becomes unavailable. Server is flooded by large number of GET or POST requests. POST requests consist of several parameters related to expensive computing on the server (like: accessing database). Hence, HTTP-POST request flooding becomes more destructive than HTTP-GET request flooding. Attackers generate a number of botnets that eventually generate a large number of http requests towards server similar to legitimate users and hence it's difficult to differentiate attack traffic from normal traffic.

Http flooding attack can be customized to enhance its effectiveness by adding SQL injection attack with it. Due to the defective coding, application can be unsecure and vulnerable to SQL inject attack. This kind of requests run database query and a large amount of queries can consume considerable amount of resources to disrupt the server's functionalities.

Http slow request attack: Several DDoS attack tools generate http slow request attack by sending high workload requests within single http session. Attackers use non-spoofed IP address to send valid packets that hold an http session for long time. It sends all http traffic in tiny fragments so slowly that the http session timeout is just not over. Server waits to receive all the fragments during this long lasting session and finally it becomes congested by multiple long lasting sessions generated by botnets. Slowloris or Slow HTTP GET attack is launched in this way by repeatedly transmitting small amount of data. Another variation of this scenario, Slow HTTP POST or slow body [21] attack sends POST request parameters and relative values without reaching the Content-Length limit. The botnet repeatedly looks for wait timeout value and sends another randomly generated POST request and corresponding values to elongate the session.

Http slow read attack: Http slow read attack affects the core application part of lower layers (for example: TCP) and makes it reply slowly. Attacker's machine or other compromised machines set the receiving window smaller than the victim server's send buffer. Hence TCP maintains open connections even if there is no data communication that eventually causes a DDoS flooding attack.

5. DDoS Defense Strategy

Most of the DDoS attacks are cumulative in nature and become more and more destructive in course of time. Any DDoS defense system aims to detect the attack as early as possible and to mitigate it as near as possible to the attack sources. Though it is expected to diminish the attack near to the source, the accuracy of detection and response mechanism at that location cannot be unquestionable. Here we will discuss some of the existing mechanism proposed in literature to defend the cloud environment from DDoS attack.

Several defense strategies have been discussed in literature for mitigating DDoS attack at application level and network level [47, 48]. In case of network level attack it is easier to detect and mitigate the attack as compared to application

level attack. IP address spoofing is a big annoyance in cloud security infrastructure as it leads to false detection of attack source.

Ingress/Egress filtering mechanism helps to distinguish spoofed IP address from the legitimate one that stays within the range of the valid addresses. Ping Du et al. [43] proposed a "Network Egress and Ingress Filtering" that can be installed at the border routers of the ISPs so that DDoS attack from the ISP and towards ISP can be mitigated. Large flows that require resource more than threshold limit are mainly accountable for DDoS attack. This filtering makes those flows restricted to limited resources. However ingress/egress filtering may not detect spoofed IP address that attackers keep within valid IP address range.

Abraham Yaar et al. [44] proposed a packet marking mechanism namely "path identifier" where packets are embedded with path fingerprint that allows users to trace back the packet transmit path from victim to source despite address spoofing. Ruiliang Chen [45] et al. also presented "Router Interface Marking (RIM)" scheme that performs packet marking with "routers interface identifier" to detect attack source by IP trace back. Trace back mechanism proposed by Yang Xiang et al. [46] can trace the attack source upto its local administrative network with lower computation cost and higher accuracy. However deployment of trace back mechanism needs the routers in that network that support identifiers to trace back. Also attackers can generate and forge the trace back message to bypass this defense strategy. Total operational costs for implementing trace back mechanisms need to be considered.

Hop count filtering method [49, 50, 51] is also a DDoS detection strategy that can differentiate spoofed packets from legitimate users' packets. As the packet travels through a route, each intermediate router decreases the TTL value of the packet by one and hence TTL value implicitly indicates the hop count between source and destination. Hop count of packets in normal traffic is calculated and stored in a table. During the attack period the hop count value is calculated for each IP address and compared with corresponding saved values. A high discrepancy between these two values makes the system discarding the packets. If machines from legitimate users' range and valid hop count are compromised by, the system may become ineffective for anomaly detection.

In traffic level measurement based defense system, a particular limit of incoming traffic is set and system throttles the traffic flow by discarding packets when congestion reaches beyond the predefined limit. Yoohwan Kim et al. [52] proposed an anomaly detection and congestion control mechanism "Packetscore" [53] for statistical packet filtering. Legitimate packets are estimated using "Conditional Legitimate Probability (CLP)" and malicious packets are discarded selectively for controlling the overload.

Existing defense strategies are broadly divided in three types based upon the deployment location of mitigation techniques, such as: source (attack -source) based approach, network or router based approach and host (destination) based approach. In cloud environment, service providers keep the data duplicated in several data centers which are geographically distributed throughout the globe. This data

Table 1. Surviving Techniques

Features	Advantages	Limitations
Chi-Chun Lo et al. [56] proposed a cooperative intrusion detection system (IDS) framework.	Cooperative agents from IDS deployed in each cloud environment exchange alerts if one IDS identifies any attack. Alerts coming from different regions are collected by alert clustering module and decision about accepting the alert is taken based upon severity of the attack. This system protects cloud environment from single point of failure.	Implementation of the cooperative agent and the majority voting system include much computational effort to existing defense system. Eventually, the system can experience high computation time and low detection rate of attacks. Also, to build this model, special cloud infrastructure is needed.
AmanBakshi et al. [57] proposed IT virtualization strategy to secure cloud environment from DDoS attack	SNORT [58] like IDS in virtual machines is used to analyze incoming and outgoing packets and to evaluate with known signature. If DDoS attack is detected, target application is shifted to other virtual machine at different data centre and packets from malicious IP addresses are blocked. This approach prevents DDoS attack in virtualized cloud environment by securing applications running in virtual machines.	SNORT kind of IDS identifies known attacks; hence all kind of DDoS attacks are not detected and prevented in virtualized environment.
N. Jeyanthi et al. [59] proposed Packet Resonance Strategy (PRS) to detect and prevent DDoS attack from Spoofed addresses	PRS implements a defense mechanism consisting of two levels: packet bounce and packet transit. It permits access to cloud datacenters only if remote clients satisfies initial authentication at both the levels. This light weight solution is able to detect malicious packets from spoofed addresses and discards those packets at DC's firewall.	It obtains intended communication channel for authenticated users. Also tracing the attack source to block further traffic flow from those addresses was not discussed
P. Arun Raj Kumar et al. [60] proposed Neural Classifier for detecting DDoS attack	"Resilient Back Propagation (RBP)" was selected as base classifier for collecting network traffic data, processing and classifying the attack. This method provides high detection accuracy with less false positive results.	The system accepts a collection of classifier outputs and "Neyman Pearson cost minimization strategy" for attack classification. It increases computational overhead for overall performance.
Chen Qi et al. [61] proposed a confidence based filtering (CBF) method for mitigating DDoS attack	CBF is a packet filtering method that generates a nominal profile for normal, legitimate packets during non-attack period and evaluates the score of packets during attack period to decide if the packet can be discarded or not. This allows dynamic packet filtering with high accuracy in very less time.	This method scores the packets based on some characteristics concurrently appeared in legitimate packets. But specific number of single attributes are not defined that need to be selected. To accumulate the confidence values of attribute-value pairs, a database is maintained at server side to store them in a 3-dimensional array due to which computing speed can be affected.
SampadaChavan et al. [62] proposed a neuro-fuzzy based intrusion detection system	An Artificial Neural Networks and Fuzzy Inference System based defense mechanism that uses SNORT for real time traffic analysis. Signature pattern database is built from supervised and unsupervised learning method.	Significant training time can restrict it to be used in a dynamic network.
Kleber Vieira et al. [63] proposed a neural network based anomaly detection scheme in grid and cloud computing	An Artificial Neural Network based anomaly detection mechanism having an audit system to secure the cloud from attacks.	It cannot work efficiently if training data is limited. Also intrusion detection takes much time.
N. Jeyanthi et al. [64] proposed an entropy based DDoS attack prevention approach	It analyzes dynamic traffic data, detects deviation of traffic from normal behavior and distinguishes normal traffic and attack traffic.	Application based attack and source address spoofing can bypass this security system.

redundancy helps to switch to other data center if one data centers experiences a high volume DDoS attack and thus to maintain the continuity of the service to legitimate users [55]. This process is suitable for large scale data centers, but for medium to small scale service providers, scope of data redundancy may be limited and a DDoS attack causes substantial loss for them. Table1 gives some of the existing surviving techniques in Cloud environment and their features have been discussed.

There are also existing works in literature where Fuzzy logic

based rules and deduction system has been studied and implemented to approach the anomaly detection mechanism. In most cases, fuzzy based intrusion detection systems suffer from limited attributes of data collection explicitly for a specific kind of attack. From design point of view, the distributed and collaborating nature of network and cloud environment has made the task more difficult. Some of the detection systems consider only distributed architecture and able to detect the attacks locally [54]. For local detection, each distributed component is able to detect anomaly locally

for that node only and aware of the local phenomenon. However global alarm is not always raised to defend large volume of attack towards the system. So the choice of detection parameters should be hybrid and anomaly based in distributed architecture. Below table shows a comparative study on some of the techniques:

Table 2.Fuzzy Based Techniques

Papers	Features	Limitation
Type-2 fuzzy set based algorithm for detecting misbehaving nodes [68]	Collects sample data of various network parameters in distributed environment for partial-anomaly based detection from misbehaving nodes.	Routing protocol is not specified, simulation result is not given
Fuzzy Logic Controller based IDS [69]	Collects audit log file and neighbors related data for misuse based detection	Specific for false route request attack
Energy based trust solution using fuzzy logic for anomaly detection [70]	Uses network packet data as source for anomaly detection	Not collaborative and response system is not given
Fuzzy Sets based Agent communication [71]	Collect packet data from data stream for misused based detection; independent and collaborative	Routing protocol is not specified, prevention scheme could be presented
Trust and fuzzy logic based detection system [72]	Uses network level data for cryptographic algorithm and trust based anomaly detection.	Only malicious node detection in collaborative way
Forensic analysis based on fuzzy logic approach [73]	Exploitation detection from Data packets and Routing packets	Not collaborative, prevention scheme and simulation are not specified
Fuzzy inference system based anomaly detection [74]	Specification and anomaly based detection from Data packets and Control packet based features	Restricted to blackhole attack only
Fuzzy logic based forensic analysis [75]	Forensic analysis based upon data stored in log files, configuration settings, routing tables etc.	Limited to small level attack

6. Fuzzy Based Defense Mechanism

We will discuss the design of a hybrid fuzzy defense mechanism against DDoS attack based on the statistical behavior of parameters of network protocols. We plan to consider parameters from network level as well as application level protocols that would help to depict the traffic pattern in a data center server. DDoS is not a single kind of network attack but a general name of different kinds of attack strategies that exploit the loopholes in existing security systems and protocols to disrupt the victim's resources. We would select the vital network parameters that change significantly during an attack phase and hence its

pattern gives an essential clue to detect denial of service attack from normal traffic.

Before launching the attack, an attacker sends ICMP Echo packets to find the machines which are vulnerable to security threat and gains their access. Once those machines are compromised, those become the agents to consolidate a DDoS attack towards a single destination. Lee et al. [10] described several network parameters that can be used to detect the DDoS parameters. Distribution of Source and Destination IP addresses and ports in existing network provide information about the DDoS attack. During the attack period the destination IP address becomes common in each packet trace. The self-similarity of each network that exists regardless of network type, protocols, topology and packet size plays a crucial role in statistical anomaly detection.

The parameters which were considered by Lee et al [10] to detect anomaly are as follows:

Source IP address and port

Destination IP address and port

Packet type

Occurrence rate of packet type (TCP SYN, UDP, ICMP)

Number of packets

Significant divergence of these parameters shows the attack in network traffic. This divergence can be measured by the concept of entropy [11] as it depicts the randomness or uncertainty of information. Shannon's theorem [12] shows that if an information source is having n independent symbols each with a probability of choice P_i then entropy H would be:

$$H = -\sum_{i=1}^n P_i \log_2 P_i \quad (1)$$

Entropies are calculated on sample of packet headers that helps to decipher the change in the pattern. During the attack traffic, the entropy of source IP addresses increases as number of sources from where the packets arrive is large and the entropy of destination IP addresses is converged to a small value as the attack is concentrated towards a particular server. In normal traffic, the distribution of source IP address of legitimate users can be seen uniformly scattered across the network. On the contrary, the distribution pattern of malicious IP addresses appears to be cumulative in specific zones like clusters. The reason behind generating such pattern is due to a number of machines in the same LAN or WAN get compromised and become agents to launch distributed attack (38). Entropy of packet type and packet rate are also considered as in earlier discussion we have seen that different DDoS tools and scripts use specific type of packets (for example: ICMP packet, UDP packet) which results into small entropy value during attack phase. Value of number of packets becomes very large during volumetric attack and its entropy also becomes large. Hence entropy based defense mechanism dynamically analyze and distinguish the malicious and normal packets according to traffic behavior that provides effective solution against DDoS attack [49].

For our fuzzy based hybrid defense mechanism we would consider some of the above mentioned parameters along with some http packet parameters so that the system can detect layer 3 attacks and can show the possibility of layer 7 attack

if needed. The parameters to be considered for DDoS detection in our proposed system can be summarized as:

Entropy of Source IP address and port
Entropy of Packet type
Packet rate (packet flow per unit time)
Number of packets
Http request rate (GET)
Http packet timeline

Secondary parameter:

Destination IP address and port

We have kept the measurement of entropy of destination IP and port as secondary parameter as it can be taken as an optional parameter. During the attack period this measurement becomes almost unique and seldom changes. So if the detection system works at boundary routers of the cloud data centers, the entropy value of Destination IP address and port will not portray much useful information. As shown in [13], packet rate can be calculated as:

$$R_t(TCP \vee UDP \vee ICMP)_i = \frac{\text{Totalnumberofincomingpackets}(TCP \vee UDP \vee ICMP)}{\text{TotalnumberofIPpackets}}$$

$$R_t(TCP \vee UDP \vee ICMP)_o = \frac{\text{Totalnumberofincomingpackets}(TCP \vee UDP \vee ICMP)}{\text{TotalnumberofIPpackets}}$$

If the http request data is abnormally small it may be the reason of slow read packets. Http packet timeline is important to consider for mitigating slow http request attack. Attacker machine requests with extremely slow packet transfer rate that keeps the server's resources always busy. Also the pattern of Http GET request is a common attribute that is present in http GET flood attacks [16]. Considering http GET request (GET request per second) [19]

The proposed fuzzy model is a similarity based learning mechanism to detect the attack traffic. Self-similarity characteristic of attack and normal traffic are typically different. The degree of similarity between these two elements can be calculated by Hurst Parameter. As shown in [14]:

$P_i : i = 1, 2, \dots, n$ represents the values of n successive values of a parameter P in incoming traffic, k is the observation time, n is the total number of observations, then expectation of X_k is defined as:

$$\bar{P}_k = \frac{1}{n} \sum_{k=1}^n P \quad (2)$$

$$\text{Mean deviation} = P_k - \bar{P}_k \quad (3)$$

The minimum and maximum deviations are defined as:

$$\min_{k \leq n} (P_k - \bar{P}_k) \quad (4)$$

and

$$\max_{k \leq n} (P_k - \bar{P}_k) \quad (5)$$

The range of n successive values:

$$R_n = \min_{k \leq n} (P_k - \bar{P}_k) - \max_{k \leq n} (P_k - \bar{P}_k) \quad (6)$$

$$\text{Normalized constant} = Q_n = R_n / S_n$$

Where Standard Deviation is defined as,

$$S_n = \sqrt{\frac{1}{n} \sum_{k=1}^n P_k^2 - \left(\frac{1}{n} \sum_{k=1}^n P_k \right)^2} \quad (7)$$

If the value of is very large then:

$$R_n / S_n \approx cn^H \quad (8)$$

Where c is a constant and H is the Hurst Parameter.

In detection phase of DDoS attack, the Traffic Class is defined by proposed fuzzy system that concludes if the traffic is normal or attack traffic. Traffic Class is the output parameter for given network parameters as input.

Problem definition: As given in [15], we are having network parameters as n input variables $x = (x_1, \dots, x_n)^T \in R_n$ and output parameter y . The sampling dataset is defined as:

$$(x^{(p)}, y^{(p)}), p = 1, 2, \dots, N$$

The fuzzy system will extract the IF-THEN rules to define the relationship between input parameters set and output parameter.

$$\text{IF } x_{i1} \text{ is } A_{i1}^{(l)} \wedge \dots \wedge x_{i3} \text{ is } A_{i3}^{(l)}, \text{ THEN } y \text{ is } B^{(l)} \quad (9)$$

Here $A_{il}^{(l)}$ and $B^{(l)}$ are the fuzzy sets, $l = 1, 2, \dots, M$ is the rule index with $m < n$. The reason behind keeping number of m less than n is that we need not choose all the n variables to define each rule but only selected number of parameters m from n available variables. The aim of the design is to find the fuzziness of output variable when a set of input conditions (variables) are defined for some fuzzy regions. Output variable would be selected from some predefined fuzzy sets (For example: Network Traffic is NORMAL or HIGH_ATTACK or VERY_HIGH_ATTACK etc.).

$A_{il}^{(l)}$ is the membership function which is already defined. The fuzzy set would be characterized by this membership or characteristic function. Membership function is associated with every point in the set of x . It defines the *degree of membership* of each element. Fuzzy sets have no well-defined boundaries and transition of membership function is gradual. Hence, grades of membership do not hold only 0 and 1 value but any values between 0 and 1 that represents a partial membership.

First we would compute the weighted vector for input-output dataset $(x^{(p)}, y^{(p)})$:

$$W_v^{(p)} = \prod_{j=1}^m \mu_{A_{ij}}(x_{ij}^{(p)}) \quad (10)$$

If $\sum_{p=1}^N W_v^{(p)} = 0$ then no rule can be formed. Otherwise, the weighted average can be calculated as:

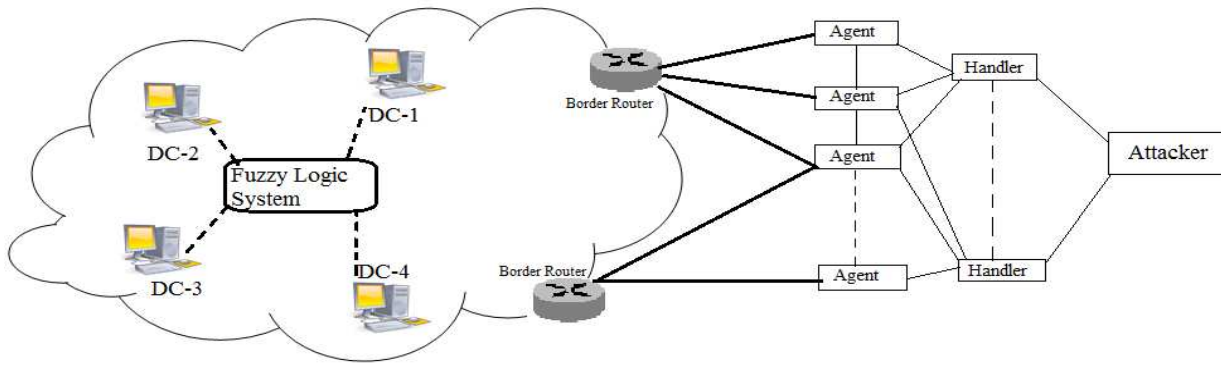


Figure 1. Fuzzy based defense system in Cloud environment

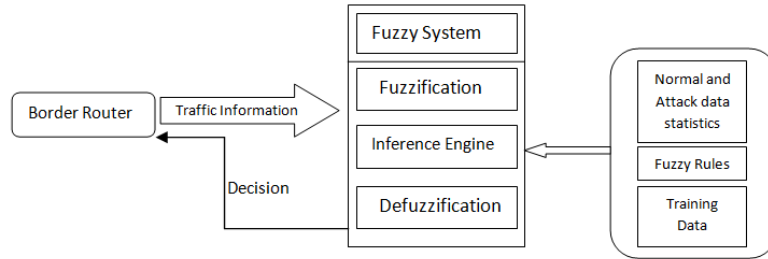


Figure 2. Detailed Design

$$w_{av} = \frac{\sum_{p=1}^N y^{(p)} w_v^{(p)}}{\sum_{p=1}^N w_v^{(p)}} \quad (11)$$

The Fuzzy IF-THEN rules would be generated from determined set of output parameters. For defined fuzzy set B^1, \dots, B^K there would be B^{j*} such that,

$$\mu_{B^{j*}}(w_{av}) \geq \mu_{B^j}(w_{av}) \quad \text{for } j = 1, 2, \dots, K. \quad (12)$$

Next, Degree of Confidence would be derived from:

$$doc = \left(1 - \frac{\sigma}{\max_{p,q=1}^N |y^{(p)} - y^{(q)}|}\right) \mu_{B^{j*}}(w_{av}) \quad (13)$$

7. System Architecture

Schematic model of the proposed system has been shown below. Figure.1 shows the fuzzy based defense system installed in cloud environment. The attackers spoof their machine's address and scan the network to find vulnerable computers. Once few machines are compromised, attackers gain the access right and use the handlers to intrude other network connected machines which are called 'agents'. The scanning, exploiting and compromising process is embedded in a worm program [4] that spreads into machines automatically, installs itself in the machines and launch further attack. Hence these agents perform the DDoS attack by bombarding large number of packets simultaneously to a target system.

The worm program generates botnets faster and the path of source is hard to trace back.

As shown in the Figure 1, the agents make attack towards the cloud server. The fuzzy intelligent system is installed in the cloud environment that makes decision out of incoming traffic to detect the DDoS attack.

Working phases: The working phases of the system can be divided into four:

Learning Phase: In this phase the inference rules are designed and fed to fuzzy systems. First the required parameters or inputs to the system are declared. These parameters are the packet characteristics that change considerably during the DDoS attack. Fuzzy system learns to make decision based upon data fed and determines the traffic class.

Traffic Analysis: In this phase, the fuzzy based defense system monitor the traffic dynamically, analyzes and evaluates the traffic class based upon inference rules. The fuzzy rules are defined in conditional way in IF-ELSE form to determine the logic. Here the rules for defending DDoS attack are flexible and can be modified based upon type of attack and the network parameters change due to the attack. Rules can be defined as:

{IF entropy of source IP and port are LOW and ICMP packet rate is HIGH THEN Traffic Class is attack HIGH
 IF Http Packet Rate is HIGH and Http Packet timeline is LOW THEN Traffic Class is attack HIGH
 IF entropy of source IP and port are HIGH and Number of packets is LOW THEN Traffic Class is NORMAL
 IF entropy of source IP and port are LOW and Number of packets is LOW and UDP packet rate is HIGH THEN Traffic Class is attack HIGH
 IF entropy of source IP is LOW and source port is HIGH and Number of packets is HIGH and Entropy of packet type is HIGH THEN Traffic Class is attack HIGH
 IF entropy of source IP and port are MEDIUM and Number of packets is LOW THEN Traffic Class is NORMAL
 IF entropy of source IP and port are MEDIUM and Packets rate of TCP SYN is HIGH THEN Traffic Class is ATTACK HIGH
 IF entropy of source IP and port are HIGH and Packets rate of TCP SYN is HIGH and Number of Packets is HIGH THEN Traffic Class is ATTACK HIGH}

Membership function for each input parameter is defined as shown in below example:

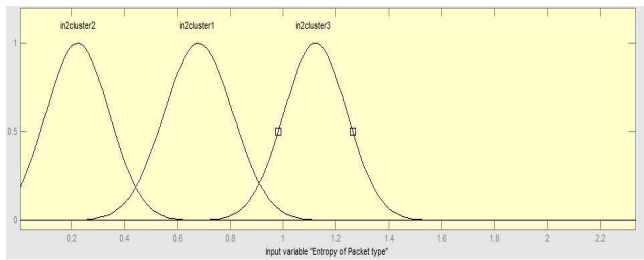


Figure 3.Membership Function plot

In this way the cloud security admin can define the fuzzy rules for possible attack types and network parameters.

Anomaly Detection: Fuzzy system determines the traffic class and generates alarms if anomaly is found.

Attack Prevention: Border routers are asked to discard the packets from malicious sources.

Distinguish DDoS from flash crowd: Distinguishing a DDoS attack from flash crowd [65] and shrew attacks [67] is a difficult job that any DDoS defense system should take care of. These are legitimate traffic patterns that create sudden surge in network packet flow when a large number of valid users try to access the service concurrently. These events do not reach any harm to data centers and does not stay for long period of time. In our solution, the defense mechanism monitors the source IP distribution and packet characteristics to discriminate a flash crowd traffic and actual DDoS attack.

8. Experimental Setup

Overall Traffic flow: We tested our proposed solution in a simulation environment. An experimental cloud data center consisting of number of virtual machines was aimed for network and application based flooding attack from multiple sources. The attack traffic was manipulated and replicated to target the victim's computing resources such as CPU, bandwidth, memory etc. The experimental results of attack traffic without the proposed defense system and with proposed defense system have been shown below.

A sudden surge in incoming traffic was experienced at border router of test data center when the flooding attack was launched.

Figure 4 shows a part of network traffic statistic at the DC during the DDoS flood attack. The graph shows the packet flow with X-axis as time (second) and Y-axis as number of packets/second.

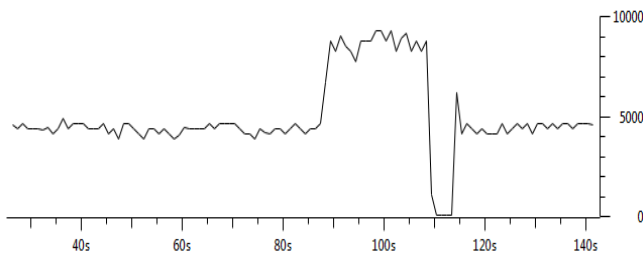


Figure 4. Traffic approaching DC during attack

Figure 5 shows the network traffic statistic approaching the data center when the DC is equipped with proposed DDoS defense system. There has been a significant packet drop at border router while mostly legitimate packets are passed to avail the service from cloud environment.

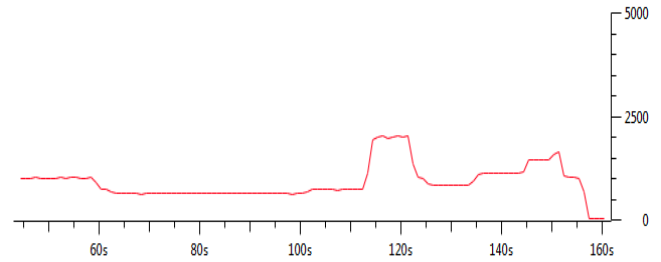


Figure 5. Attack traffic approaching DC with defense system

Figure 6 shows the normal traffic flow when there is no DDoS attack. Hence the comparison can be seen here how a DDoS flood attack increases the network traffic substantially in cloud environment while legitimate users cannot get the service hosted in DC.

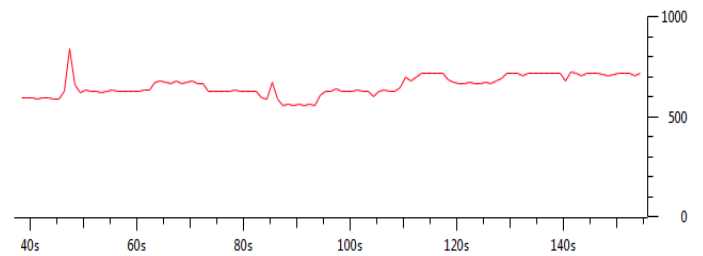


Figure 6.Traffic approaching DC when no attack

The experimental result of the defense system is shown below:

Table 3.Experimental Result

Network Class	Number of test data	Accurate detection	Inaccurate detection
Attack Low	2650	2436	214
Attack Medium-High	2800	2434	366
Attack High	3100	2990	110
Normal	4200	4106	94

We would calculate the accuracy of the system with performance measurement matrices as defined in [47]:

Table 4.Result

Attack Medium-High		Expected detection	
		Negative	Positive
Actual Traffic Class detection	Negative	A=1009	B=142
	Positive	C=224	D=1425

Accuracy = $(1009+1425) / (1009+142+224+1425) = 0.8693$

Sensitivity = $1425 / (142+1425) = 0.9094$

Specificity = $1009 / (1009+224) = 0.8183$

Precision = 0.8642

False Positive Rate (Reliability) = 0.1358

False Negative Rate = 0.1234

Email Server Response Time: Email response time by the email server is a measure of time elapsed between sending and receiving email requests. When the email server in cloud

is affected by DDoS attack, a significant delay in email response time is experienced. Figure 7 shows the email response time captured in three different scenarios and a significant drop in attack packets is seen while defense mechanism is on during attack.

FTP Server response time: Same as email server response time, FTP server response time is also affected considerably. The elapsed time is measured by duration between sending an ftp packet request to server and receiving the packet from server. While the server experiences DDOS attack, the ftp packet response is delayed. Figure 8 shows different scenarios of ftp server response time under attack and without attack.

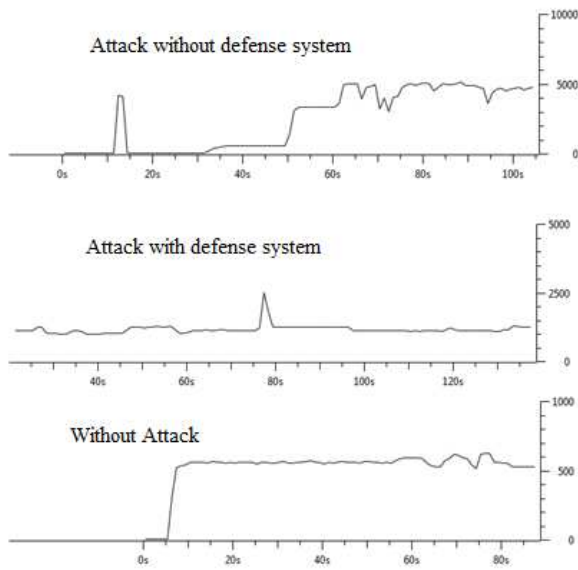


Figure 7. Traffic approaching email server

Profit Analysis: As discussed in [67], cloud operational cost is estimated based upon data transmission and memory R/W operations at data centers. If T = time (hrs), C_{BW} = Bandwidth Cost, C_{MEM} = physical memory cost, C_{VM} = VM environment cost, C_{DS} = data storage cost, then total cost at data center can be calculated as:

$$Total\ Cost = \sum_{i=1}^N \{C_{BW} + C_{MEM} + C_{VM} + C_{DS}\}$$

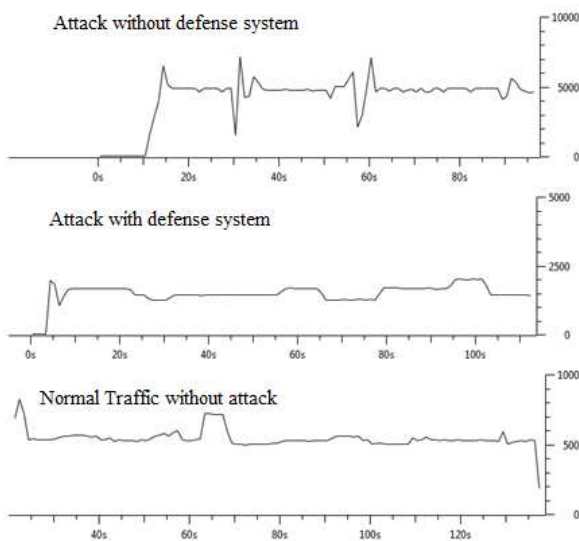


Figure 8. Traffic approaching ftp server

DDoS defense mechanism provides resource protection and resource availability to legitimate users. It detects attackers early and prevents them to avail further access of the cloud data centers. Data transmission and memory operation cost are reduced considerably. Hence it contributes significantly in saving cost at data centers. The cost may vary based upon individual invested on each constraint.

Comparison with existing schemes: The fuzzy logic based scheme described in this paper is an anomaly and pattern based detection technique that considers DDoS attack in Cloud Environment from multiple aspects. As the fuzzy rules can be deduced from various key network parameters whose values are mostly altered in a DDoS scenario, a service provider can get the flexibility of defining own rules as per the services offered. Unlike existing schemes as discussed earlier, this fuzzy based defense mechanism provides a hybrid and cooperative, simpler yet robust system, effective for detecting and defending DDoS attack in a distributed, cloud network from multiple perspectives with a proven experimental outcome.

9. Conclusion and Future Work

DDoS is a common yet powerful attack that exhausts the computing resources of the data centers and disrupts the overall service hosted there. With the advent of different DDoS attack tools, flooding from distributed sources has become easy to launch and difficult to mitigate. It is also required to distinguish legitimate packets from attack packets so that valid users are not affected by the attack and they get uninterrupted service. In this paper we have proposed a fuzzy logic based defense mechanism that is first trained with training data and rules are defined as per the possible traffic pattern of the cloud environment so that the system can infer the traffic class based upon acquired knowledge. We have taken some predefined traffic parameters that vary significantly between a normal traffic pattern and attack traffic pattern and defined the fuzzy rules based upon that. However for any particular data center, from ddos traffic pattern, the parameters can be changed based upon specific requirements and rules can be designed based upon that. Day by day the attackers are coming up with more sophisticated ways of generating DDoS attack. So there cannot be any ultimate defense mechanism that can protect DCs from all kind of DDoS attack. Hence defending approaches should also be updated and modified frequently. Our future work is to design the DDoS defense mechanism with more fine-tuned intelligent, knowledge based system so that it can defend sophisticated attacks further in the application level.

References

- [1] C. Michael, C.Silverman. Mafiaboy: "How I Cracked the Internet and why It's Still Broken," ISBN: 978-0670067480, Renouf Pub Co Ltd, First Edition, 2008.
- [2] C.Czosseck, Rain Ottis, T. Anna-Maria, "Estonia after the 2007 cyber attacks: Legal, strategic and organisational changes in cyber security," International Journal of Cyber Warfare and Terrorism, Vol. 1, No. 1, pp. 24-34, 2011.

- [3] N. Chantler, "Profile of a Computer Hacker," ISBN: 978-0-96287-002-6, Interpact Press Publishers, Illustrated Edition, 1997.
- [4] Stacy Prowell, Rob Kraus, Mike Borkin, "Seven Deadliest Network Attacks," ISBN: 978-1-59749-549-3, Syngress Publishers, First Edition, 2010.
- [5] Lin, Shun-Chieh, Shian-Shyong Tseng, "Constructing detection knowledge for DDoS intrusion tolerance," *An International Journal of Expert Systems with applications*, Vol. 27, No. 3, pp. 379-390, 2004.
- [6] Leland, Will E., Murad S. Taqqu, Walter Willinger, and Daniel V. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Transactions on Networking*, Vol. 2, No. 1, pp. 1-15, 1994.
- [7] Nazario, Jose, "Politically motivated denial of service attacks," *The Virtual Battlefield: Perspectives on Cyber Warfare*, ed. Christian Czosseck and Kenneth Geers (Amsterdam: IOS Press,) pp. 163-181, 2009.
- [8] V.Segura, and J. Lahuerta. "Modeling the economic incentives of ddos attacks: Femtocell case study," *In Economics of Information Security and Privacy*, Springer US Publishers, pp. 107-119, 2010.
- [9] S. Ranjan, R. Swaminathan, M. Uysal, E. Knightly, "DDoS-Resilient Scheduling to Counter Application Layer Attacks under Imperfect Detection," 25th IEEE International Conference on Computer Communications, Barcelona, pp. 1-13, 2006.
- [10] K.Lee, J. Kim, Ki Hoon Kwon, Y. Han, S. Kim. "DDoS attack detection method using cluster analysis," *An International Journal of Expert Systems with Applications*, Elsevier, Vol. 34, No. 3, pp. 1659-1665, 2008.
- [11] L.Feinstein, D.Schnackenberg, R. Balupari, D. Kindred, "Statistical approaches to DDoS attack detection and response," *In DARPA Information Survivability Conference and Exposition*, Washington DC, Vol. 1, pp. 303-314, 2003.
- [12] C.E.Shannon, W.Weaver, "The mathematical theory of communication," Urbana : University of Illinois Press, 1963, ©1949.
- [13] I.Ogechi, I.Chukwugoziem, H.C.Inyama. "Fuzzy modelling of a network Denial of Service (DoS) attack phenomenon," *International Journal of Engineering & Technology*, Vol. 5, No. 2, 2013.
- [14] G.W.Sikazwe. "Statistical Self-Similarity: Fractional Brownian Motion," *Time Series Seminar-March 10, 2010*, Lappeenranta University of Technology, pp. 1-22
- [15] LX Wang, "The WM method completed: a flexible fuzzy system approach to data mining," *IEEE Transactions on Fuzzy Systems*, Vol. 11, No. 6, pp. 768-782, 2003.
- [16] J. Choi, C. Choi, Byeongkyu Ko, D. Choi, P. Kim. "Detecting Web based DDoS Attack using MapReduce operations in Cloud Computing Environment," *Journal of Internet Services and Information Security*, Vol. 3, no. 3/4, pp. 28-37, 2013.
- [17] T. Peng, C. Leckie, K. Ramamohanarao, "Survey of network-based defense mechanisms countering the DoS and DDoS problems," *ACM Computing Survey*, Vol. 39, No. 1, Article 3, pp. 1-42, 2007.
- [18] D van der Sanden, "Detecting UDP attacks in high speed networks using packet symmetry with only flow data," Master thesis, University of Twente, Enschede – The Netherlands, 2008.
- [19] Wei Wei, Feng Chen, Yingjie Xia, Guang Jin, "A Rank Correlation Based Detection against Distributed Reflection DoS Attacks." *Communications Letters IEEE*, Vol. 17, No. 1, pp. 173-175, 2013.
- [20] T.Yatagai, T. Isohara, Sasase, Iwao "Detection of HTTP-GET flood attack based on analysis of page access behavior." *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, Victoria, pp. 232-235, 2007.
- [21] E.Cambiaso, G.Papaleo, M.Aiello, "Taxonomy of slow DoS attacks to web applications," *Recent Trends in Computer Networks and Distributed Systems Security*, Springer Berlin Heidelberg, Trivandrum, pp. 195-204, 2012.
- [22] G.Perry, Minimizing public cloud disruptions, TechTarget, [online]. Available at: <http://searchdatacenter.techtarget.com/tip/Minimizing-public-cloud-disruptions>, 2011.
- [23] Available at: <http://staff.washington.edu/dittrich/misc/trinoo.analysis>
- [24] NSFOCUS Mid-Year DDoS Threat Report 2013 Details DDoS Attack Trends, available at: http://en.nsfocus.com/2013/news_0912/144.html.
- [25] Denial-of-service attack cripples Microsoft for second day, available at: <http://www.networkworld.com/news/2001/0125mshacked.html>
- [26] Powerful Attack Cripples Internet [Scary], available at: http://www.greenspun.com/bboard/q-and-a-fetch-msg.tcl?msg_id=00A7G7.
- [27] SCO Sets Bounty for Worm Writer, available at: <http://www.pcworld.com/article/114479/article.html>
- [28] Answers about recent DDoS attack on Spamhaus, available at: <http://www.spamhaus.org/news/article/695/answers-about-recent-ddos-attack-on-spamhaus>.
- [29] Technical Details Behind a 400Gbps NTP Amplification DDoS Attack, available at: <http://blog.cloudflare.com/technical-details-behind-a-400gbps-ntp-amplification-ddos-attack>
- [30] Intrusion Detection FAQ: Distributed Denial of Service Attack Tools: trinoo and wintrinoo. Available at: <http://www.sans.org/security-resources/idfaq/trinoo.php>
- [31] The DoS Project's "trinoo" distributed denial of service attack tool. Available at: <http://staff.washington.edu/dittrich/misc/trinoo.analysis.txt>
- [32] NewEraCracker / LOIC, available at: <https://github.com/NewEraCracker/LOIC/>
- [33] LOIC web version with HiveMind, available at: <https://code.google.com/p/lowc/>
- [34] R-U-Dead-Yet, available at: <https://code.google.com/p/r-u-dead-yet/>
- [35] Available at: <http://security.radware.com/knowledge-center/DDoSedia/rudy-r-u-dead-yet/>

- [36] DdoSPedia,: <http://security.radware.com/knowledge-center/DDoSPedia/tors-hammer/>
- [37] Available at: <http://sourceforge.net/projects/torshammer/>
- [38] Layer 7 DDOS Attacks: Detection & Mitigation, available at: <http://resources.infosecinstitute.com/layer-7-ddos-attacks-detection-mitigation>
- [39] Kumar, Sanjeev. "Smurf-based distributed denial of service (ddos) attack amplification in internet," 2nd International Conference on Internet Monitoring and Protection, Sun Jose, pp. 25-30, 2007.
- [40] G.Kambourakis, T.Moschos, D.Geneiatakis, S.Gritzalis," Detecting DNS amplification attacks," Second International Workshop on Critical Information Infrastructures Security, Springer Berlin Heidelberg, pp. 185-196, 2008.
- [41] Defenses Against TCP SYN Flooding Attacks, available at: http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_9-4/syn_flooding_attacks.html
- [42] C.L.Schuba, Ivan V. Krsul, Markus G. Kuhn, Eugene H. Spafford, A.Sundaram, D. Zamboni, "Analysis of a denial of service attack on TCP," IEEE Symposium on Security and Privacy, Oakland, pp. 208-223, 1997.
- [43] Du, Ping, Akihiro Nakao, "DDoSdefense deployment with network egress and ingress filtering." IEEE International Conference on Communications, Cape Town, pp. 1-6, 2010.
- [44] A.Yaar, A.Perrig, D.Song, "Pi: A path identification mechanism to defend against DDos attacks," IEEE Symposium on Security and Privacy, Berkeley, pp. 93-107, 2003.
- [45] Chen, Ruiliang, Jung-Min Park, R. Marchany. "NISp1-05: RIM: Router Interface Marking for IP Trace back," IEEE Global Telecommunications Conference, San Francisco, pp. 1-5, 2006.
- [46] Xiang, Yang, Ke Li, W. Zhou, "Low-rate DDos attacks detection and traceback by using new information metrics," IEEE Transactions on Information Forensics and Security, Vol. 6, No. 2, pp. 426-437, 2011.
- [47] Zargar, T.Saman, James BD Joshi, and D. Tipper,"A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks," IEEE Communications Surveys & Tutorials, Vol. 15, No. 4, pp. 2046-2069, 2013.
- [48] Tony T.N Miu, Albert, K.T.Hui, W. L. Lee, Daniel XP Luo, Alan KL Chung, Judy WS Wong, "Universal DDos Mitigation Bypass," Black Hat USA, 2013.
- [49] C. Jin, H. Wang, Kang G. Shin, "Hop-count filtering: an effective defense against spoofed DDos traffic," 10th ACM conference on Computer and communications security, Washington, pp. 30-41, 2003.
- [50] H. Wang, Cheng Jin, Kang G. Shin, "Defense against spoofed IP traffic using hop-count filtering," IEEE/ACM Transactions on Networking (TON), Vol. 15, No. 1, pp. 40-53, 2007.
- [51] P.Varalakshmi, S. ThamaraiSelvi, "Thwarting DDos attacks in grid using information divergence," Elsevier Journal on Future Generation Computer Systems, Vol. 29, No. 1, pp. 429-441, 2013.
- [52] Y.Kim,, W. C. Lau, M.C.Chuah, H. J. Chao, "PacketScore: a statistics-based packet filtering scheme against distributed denial-of-service attacks," IEEE Transactions on Dependable and Secure Computing, Vol. 3, No. 2. pp. 141-155, 2006.
- [53] M.C. Chuah ,W.C. Lau,Y. Kim ,H.J. Chao, "Transient Performance of PacketScore for blocking DDos attacks." IEEE International Conference on Communications, Paris, Vol. 4, pp. 1892-1896, 2004.
- [54] A.Chaudhary, V. N. Tiwari, A. Kumar. "Analysis of Fuzzy Logic Based Intrusion Detection Systems in Mobile Ad Hoc Networks," BVICAM's International Journal of Information Technology, BharatiVidyapeeth's Institute of Computer Applications and Management , Vol. 6 No. 1, pp. 690-696, 2014.
- [55] K.R.Joshi, G.Bunker, F.Jahanian, A.van Moorsel, J. Weinman,"Dependability in the cloud: Challenges and opportunities," IEEE/IFIP International Conference on Dependable Systems & Networks, Lisbon, pp. 103-104, 2009.
- [56] Lo, Chi-Chun, Chun-Chieh Huang, Joy Ku. "A cooperative intrusion detection system framework for cloud computing networks," IEEE 39th International Conference on Parallel Processing Workshops , San Diego, pp. 280-284, 2010.
- [57] Bakshi, Aman, B. Yogesh, "Securing cloud from ddos attacks using intrusion detection system in virtual machine," IEEE Second International Conference on Communication Software and Networks, Singapore, pp. 260-264. 2010.
- [58] Snort@: an open source network intrusion prevention and detection system - <http://www.snort.org>
- [59] N.Jeyanthi, N.Ch.S.N.Iyengar,. "Packet Resonance Strategy: A Spoof Attack Detection and Prevention Mechanism in Cloud Computing Environment." International Journal of Communication Networks & Information Security, Vol. 4, No. 3, pp. 163-173, 2012.
- [60] R.Kumar, P. Arun, S.Selvakumar,"Distributed denial of service attack detection using an ensemble of neural classifier,"International Journal of Computer Communications, Vol. 34, No. 11, pp. 1328-1341, 2011.
- [61] Chen, Qi, W. Lin, W. Dou, Shui Yu. "CBF: A packet filtering method for DDos attack defense in cloud environment," IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing,, Sydney, pp. 427-434, 2011.
- [62] S.Chavan, K. Shah, Neha Dave, S.Mukherjee ,A.Abraham ,S.Sanyal, "Adaptive neuro-fuzzy intrusion detection systems," International Conference on Information Technology: Coding and Computing, Las Vegas, Vol. 1, pp. 70-74, 2004.
- [63] K.Vieira, A. Schulter, Carlos Westphall, Carla Westphall, "Intrusion detection for grid and cloud computing." It Professional, Vol. 12, No. 4, pp. 38-43, July 2010.
- [64] N.Jeyanthi, N.Ch.S.N.Iyengar, P.C.M.Kumar, A. Kannammal. "An Enhanced Entropy Approach to Detect and Prevent DDos in Cloud Environment," International Journal of Communication Networks & Information Security(IJCNIS), Vol. 5, No. 2, pp. 110-119, 2013.

- [65] J. Jung, B. Krishnamurthy, M. Rabinovich, "Flash Crowds and Denial-of-Service Attacks: Characterization and Implications for CDNs and Web Sites," Proceedings of the 11th international ACM conference on World Wide Web, Honolulu, pp. 293-304, 2002.
- [66] Y. Chen and K. Hwang, "Collaborative Detection and Filtering of Shrew DDoS Attacks using Spectral Analysis," Journal of Parallel and Distributed Computing, SI on Security in Grids and Distributed Systems, pp.1137-1151,2006.
- [67] N.Ch.S.N.Iyengar, G.Ganapathy. P.C.M.Kumar,A. Abraham, "A Multilevel Thrust Filtration Defending Mechanism against DDoS Attacks in Cloud Computing Environment," International Journal of Grid and Utility Computing (IJGUC), Vol. 5, No. 4, pp. 236-248, 2014.
- [68] A. Visconti, H. Tahayori, " A Biologically – Inspired type-2 fuzzy set based algorithm for detecting misbehaving nodes in ad hoc networks," International Journal for Infonomics, Vol. 3, No. 2, pp. 270-277, 2010.
- [69] S.Sujatha, P. Vivekanandan, A.Kannan, "Fuzzy logic controller based intrusion handling system for mobile adhoc networks," Asian Journal of Information Technology, Vol. 7, No. 5, pp. 175-182, 2008.
- [70] R.Vijayan, V. Mareeswari, K. Ramakrishna. "Energy based Trust solution for Detecting Selfish Nodes in MANET using Fuzzy logic," International Journal of Research & Reviews in Computer Science ,Vol. 2, No. 3, pp. 647-652, 2011.
- [71] Watkins, Damian, "Tactical Manet Attack Detection Based on Fuzzy Sets Using Agent Communication". Proceedings for the Army Science Conference (24th),Orlando, Florida, Nov 29-Dec 2, pp.1-2, 2005.
- [72] V.Manoj, Mohammed Aaqib, N. Raghavendiran, R. Vijayan, "A Novel Security Framework Using Trust and Fuzzy Logic in MANET," International Journal of Distributed and Parallel Systems (IJDPS), Vol. 3, No. 1, pp. 285-299, 2012.
- [73] S. Ahmed, S.M. Nirkhi, "A Fuzzy approach for forensic analysis of DDoS attack in manet" International Journal of Advanced Computer Science and Applications, Vol. 4, No. 6, PP.193-198,2013
- [74] D.Vydeki, R. S. Bhuvaneshwaran, "Effect of clustering in designing a fuzzy based hybrid intrusion detection system for mobile ad hoc networks," Journal of Computer Science, Vol. 9, No. 4, pp. 521-525, 2013
- [75] S. Ahmed, S. M. Nirkhi, "Fuzzy Forensic Analysis System for DDoS Attack in MANET Response Analysis," International Journal of Science and Modern Engineering (IJISME), Vol. 1, No. 7, pp. 52-55, 2013.