Identification of Trusted Elements with malicious behaviour in Public Cloud against DoS attack

Submitted By

Mishti Samani

14MCEI09

NIRMA UNIVERSITY
INSTITUTE OF TECHNOLOGY
NAAC ACCREDITED 'A' GRADE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2016

Identification of Trusted Elements with malicious behaviour in Public Cloud against DoS attack

Major Project

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology in Computer Science and Engineering

(Information and Network Security)

Submitted By

Mishti Samani

(14MCEI09)

Internal Guide

Prof. Jitendra Bhatia

Nirma University, Ahmedabad.

External Guide

Mr.Miren Karamta

BISAG, Gandhinagar

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2016

Certificate

This is to certify that the major project entitled **"Identification of Trusted Elements with malicious behaviour in Public Cloud against DoS attack"** submitted by **Mishti Samani (Roll No: 14MCEI09)**, towards the partial fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Engineering(INS) of Institute of Technology, Nirma University Ahmedabad, is the record of work carried out by him under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this project, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Prof. Jitendra Bhatia
Internal Guide,
Nirma University, Ahmedabad.

Dr Sharada Valiveti
PG Coordinator-INS,
Nirma University, Ahmedabad

Dr. Sanjay Garg
HOD-CSE,
Nirma University, Ahmedabad.

Dr P.N.Tekwani
Director,
Nirma University, Ahmedabad

Statement of Originality

I, **Mishti Samani**, Roll. No. **14MCEI09**, give undertaking that the Major Project entitled "**Identification of Trusted Elements with malicious behaviour in Public Cloud against DoS attack**" submitted by me, towards the partial fulfillment of the requirements for the degree of Master of Technology in **Computer Science & Engineering(INS)** of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made. It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

Signature of Student
Date: 16^{th} May,2016
Place: Ahmedabad

Endorsed by
Prof Jitendra Bhatia
(Signature of Guide)

Acknowledgements

First and foremost, sincere thanks to Mr.M.B.Potdar Director, BISAG, Gandhinagar. I enjoyed his vast knowledge and owe him lots of gratitude for having a profound impact on this report.

It gives me immense pleasure in expressing thanks and profound gratitude to **Mr.Miren Karamta**,BISAG,Gandhinagar for his valuable guidance and continual encouragement throughout this work. The appreciation and continual support he has imparted has been a great motivation to me in reaching a higher goal. His guidance has triggered and nourished my intellectual maturity that I will benefit from, for a long time to come.

I would like to thank my Internal Guide, **Prof.Jitendra Bhatia**,Nirma University, Ahmedabad for his valuable guidance. Throughout the Dessertation, he has given me much valuable advice on project work. Without him, this project work would never have been completed.

It gives me an immense pleasure to thank **Dr. Sanjay Garg**, Hon'ble Head of Computer Science and Engineering Department, Institute of Technology, Nirma University, Ahmedabad for his kind support and providing basic infrastructure and healthy research environment.

I would also like to thank **Prof.P.N.Tekwani**, Director, Institute of Technology, Nirma University, Ahmedabad for providing me an opportunity to get an internship at BISAG , Gandhinagar. I would like to thank my all faculty members for providing encouragement, exchanging knowledge during my post-graduate program.

I also owe my colleagues in the BISAG, special thanks for helping me on this path and for making project at BISAG more enjoyable.

- Mishti Samani

14MCEI09

Abstract

Open and distributed nature of cloud , vulnerability of internet , different limitations of cloud service models are some of key features for the attraction of various attackers. With the advancement of technology , cloud services are facing increasing amount of threats with the advent of new types of different attacks. DoS attack has severe impact on cloud environment as it is not limited to quality of service but also related to the maintenance of services. Intrusion detection in cloud has been reported as major security concern in cloud security alliance. Performance of IDS has been degraded as it faces numerous problems. Traditional IDS and IPS (Intrusion Prevention System)has been inefficient due to open nature of cloud. Different models of IDS are proposed as defense mechanism to strengthen network security and protect from different anomalies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Cloud Computing

Cloud computing is popular due to its numerous characteristic and benefits. Reduction of Cost, Scalable and flexible, Quick and Easy implementation, Reduced Maintenance cost, Quality of Service, Mobility and so on has advantages has made cloud popular in small and large scale industries. We have been dependent on cloud technologies such as Google docs, amazons storage cloud , Dropbox, skype and so on applications. Inspite of its numerous advantages it faces numerous security challenges such as security and privacy, loss of control and lack of standards. Some of the essential characteristics are :

a. Resource Pool: Computing Resources such as Processing Power, network bandwidth , memory and storage area must be in virtualized into some virtualized pool can be allocated dynamically based on end user demands.

b. On-Demand service: There is no need of any human intervention and provider to access server time and network storage.

c. Regular Service: It provides the facilities of resource monitoring , controlling , reporting usage of amount resources and this can be served to users.

d. Rapid Elasticity: Services provided to end-users are unlimited and provided based on their request.

e. Wide Network Accessibility: Services can be accessible on various devices such as mobile phones, tablets, laptops, workstations.

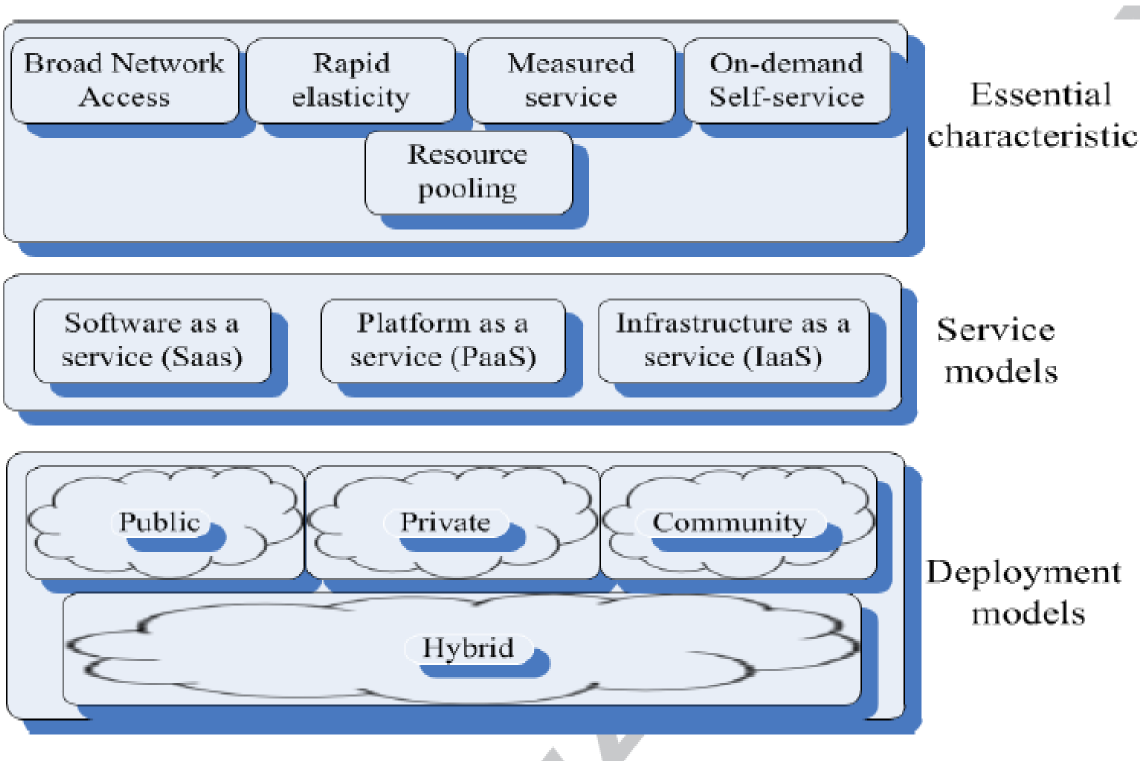

Figure 1.1: Cloud Architecture [6]

cloud computing consists of three layers:

a. **cloud application:** Its the uppermost layer which eliminates the need of installing applications and software on individual computer as it can easily be accessed by web browser, hosted desktop or remote client. This layer has following security limitations. Saas deployment model(firewall, IDS),MITM(man -in-the-middle) attacks, OS and SQL Injection Flaws,IP spoofing, port scanning, packet sniffing, Cross site scripting [XSS],Access control weaknesses , Cross site request forgery [CSRF],Cookie manipulation ,Network penetration and packet analysis,Hidden field manipulation, Insecure storage and configuration, Network Security, Session management weaknesses, Insecure SSL trust configuration, SQL Injection flaws ,Data validation ,Insecure storage ,Identity Management and Sign-on Process, Authentication weakness analysis, Insecure trust configuration.

b. **cloud platform:** Its the middle layer which performs necessary changes in server settings and configurations based on the demands. following are the challenges faced by this layer: Web Session impersonation, Phishing attacks , Social engineering and brute force attacks, Password Reset attacks,application's default configurations, SSL Protocol and implementation flaws,Insecure permissions on cloud data,cross site scripting to websites ,side channel attacks ,lack of secure SDLC(SSDLC) ,Protect private information before sending it to cloud by means of encryption,keeping an audit trail, protection of API Keys,inadequate security by cloud provider,integration with the rest of systems.

c. **cloud infrastructure:** Its the lowest layer. Its main function is to provide infrastructure using virtualization. The limitation of this layer is that the security features by CSP are not updated and as per needs. This limitations of traditional methods has evaded security breach in network policies , which have enforced to develop security oriented systems monitoring the system activities. Network and information systems are vulnerable to security attacks which leads to breach in policy and data. IDS is used to monitor the network traffic performing malicious activities results into violating security rules. Different risks are associated with cloud computing such as increased potential of insider attacks, side channel attacks.

1.1.1 Types of Intruders

Intruders may be external or internal and may have various impacts on network. This may vary from benign to serious.

Different types of Intruders are:

a. **Masquerader** An unauthorized individual who penetrates the system access controls by exploiting legitimate users account. It is likely to be outsider.

b. **Misfeasor** An authorized who misuses its privileges or unauthorized individual who accesses resources , programs or data. It is likely to be insider.

c. **Clandestine user** An user who is at superior position to evade auditing and access controls or suppress audit control. It is likely to be outsider or insider.

According to statistics , eighty percent cases observed suffers security breaches by internal intruders and they are hard to detect and prevent .Script Kiddies ,gray hat hacker , Amateur hackers, rival corporations, terrorists and even foreign government shave the motive and capability to carryout sophisticated and novel attacks against computer systems [**?**]. Intruder penetrates into legitimate user accounts and exploits it by creating security breaches. some of the examples of intrusions are : installing unauthorized tools , remote monitoring applications, use of packet sniffers to capture passwords and so on.

There are basically four steps for intrusion in an network:

a. **Prepare** Attacker tries to gain complete network configuration details such open ports , IP Addresses and Operating system vulnerabilities.

b. **Exploit** Once attacker recognize vulnerabilities, it would exploit them. It may even take multiple attempts.

c. **leave behind** Once attack is successful a back door is prepared by installing the softwares and network sniffers.

d. **clean up** Attacker cleans the left evidence by clearing logs and other information and installing modified system software.

1.1.2 INTRUSIONS IN CLOUD ENVIRONMENT

Different security attacks are performed with different motives and they corrupt the system in different ways. These vulnerabilities results in violations of different properties: Availability, Confidentiality, Integrity and Control.

Availability:

There should be violation of security policies if intended user or authorized users would not be able to access a particular system resources whenever they need to access it.

Confidentiality:

There would be breach in the security if the attacker can gain unauthorized access without the owners permission or information.

Integrity:

There would be violation in Integrity if unauthorized users or attackers changes the system state or data possessing the system or passing through the system.

Control:

An attack causes violation in access control by granting privileges to access control policies.

These intrusions can affect availability , confidentiality and integrity. They are:

a. **Insider Attack:** An authorized users tries to gain an higher privileged levels. Sometimes they may even disclose secret information of the organization . Such attacks are carried by employees of the organization.

b. **Flooding Attack:** Attacker sends huge number of packets of TCP, UDP, ICMP or mix of them by flooding the victim. Illegitimate network connection is responsible for the attacks. In cloud, VM is open to internet so there is high risk of DoS (or DDoS) through zombie [3].

   It mainly affects service availability which leads to loss of availability of resources to the intended or authorized users. It will completely exhaust hardware devices and would no longer able to carry out intended tasks.

c. **User to Root Attack:** Attacker tries to gain authorized access by sniffing the password. This would further be used to exploit vulnerabilities of root level access. Such attacks takes place when static buffer is overfilled. Some security risks such as password recovery workflows, phishing attack, key loggers etc. dont possess any standard mechanism to prevent security risks. In cloud, attacker tries to gain valid user instances via to obtain root level access of VM.

d. **Port Scanning:** Various port scanning techniques are TCP Scanning, UDP Scanning, SYN Scanning, FIN Scanning, ACK Scanning, and Window Scanning. They lists various open ports, closed ports and filter ports. Various Information such as IP address, MAC address, router, gateway filtering, firewall rules and so on can be known and can be misused.

e. **Attack on virtual machine or hypervisor:** To gain a complete control over virtual machine, the lower layer needs to be compromised. Some of popular attacks on virtual layer are BLUEPILL (2006), SubVir (2006) and DKSM are some well-known attacks on virtual layer.

   Zero Day vulnerabilities are found in VM to gain complete access. A hyperVM virtualization application was exploited by zero-day vulnerability which resulted in destruction of many server based websites (2009) [2].

f. **BackDoor Channel Attacks:** Hacker gains remote access in infected code by compromising confidentiality. Thus it is a passive attack that can control victims resources and use it as zombie to perform DDoS attack. Attacker can get access and control of cloud user resources by compromising the system. To prevent such attacks firewall, signature and anomaly based intrusion detection system is used.

1.2 Intrusion Detection System

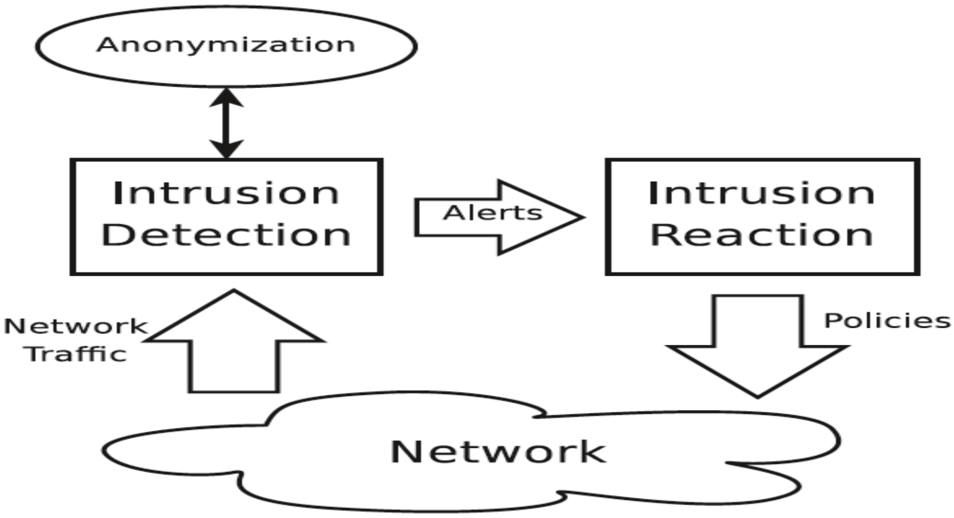

Figure 1.2: MacroComponents of IDS [14]

Three main Components of IDS are as follows:

a. **Intrusion detection system** To classify anomalous traffic summarization algorithm and pattern recognition techniques are used.

b. **Anonymizer** Some real life traces such IP addresses, application information are used to train the pattern recognition algorithm.

c. **A**lert signals acts as triggers for information exchange and trace back the attack resources.

Above architecture is dependent on classical IP Infrastructure.
Model is composed of two parts:

a. **Real time Intrusion Detection system** It is based on user behavioral model network packets are analyzed and classified.

b. **Pattern Recognition System** Data from user behavioral model is extracted and stored in database in the network traffic features and pattern recognition algorithm.

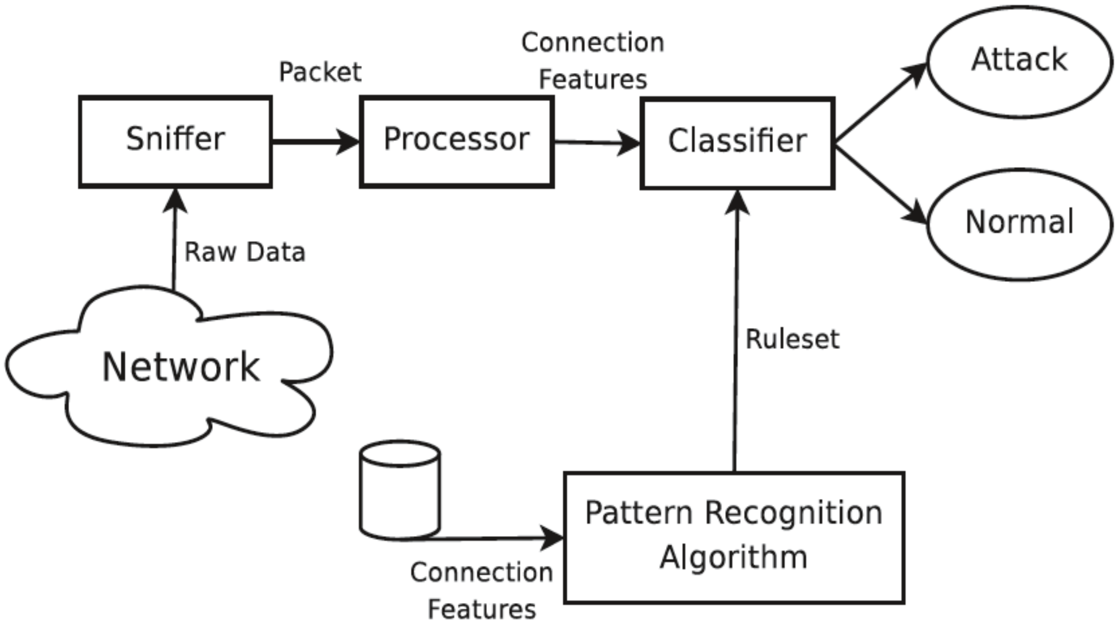

Figure 1.3: Framework for Intrusion Detection system [14]

1.3 Features of IDS

Following are the features of IDS are:

- IDS can detect the attacks , errors undergoing in the system.
- IDS can monitor the system performance automatically.
- IDS can trace users activity and their impact.
- IDS can act as support system .
- IDS can detect errors, misconfigurations along with various attacks.

1.4 existing software

1.4.1 Snort

Snort is open source Network based Intrusion Detection System that facilitates real time traffic analysis and packet logging. The main function of snort is Protocol Analysis,Content matching and Searching.

The Three main modes of Snort are:sniffer mode, packet logger mode and Network Based Intrusion Detection System. Sniffer mode is used to read network packets and display them on console. Packet logger mode is used to log packets into disk.NIDS is used to monitor network traffic and analyse based on rule set.

1.4.2 Tiger

Tiger is a open source tool used for security auditing and network intrusion detection system. Tiger can be written using shell language.Tiger checks system status and its configuration.

1.5 Need of research

According to cloud security alliance report 2015 , following are security concerns of cloud computing[22]:

- Data Breaches
- Data Loss
- DDoS Attack
- Account Hijacking
- Insufficient Due Diligence
- Malware

- Viruses
- Phishing Attacks
- BYOD

Open and distributed nature of cloud, vulnerability of internet, different limitations of cloud service models are some of key features for the attraction of various attackers.

1.6 Objective of Study

Dos attack can cause some of problems such as Ineffective services, inaccessible services, Interruption of network traffic in the connection interface. Based on the motive of attack , the attack is classified as severe. It is necessary to detect the severity of this attack based on the performance. These attack are necessary to detect so that they does not affect any of the services.

following are the ways to identify DoS attack:

a. unusually slow network performance

b. unavailability of particular site

c. increase in time span to access your account

d. inability to access website

1.7 Scope of Work

Knowledge acquisition: As I've mentioned above attack needs complete understanding of user behavior and knowledge. Network needs to be continously monitored to identify the malicious behavior. Partial understanding It is not always possible to give detect variations of attacks. So, our model must be able to overcome as many problems as possible which is the scope of the project. It is also necessary to detect attacks by minimization of false alarm rates.

Chapter 2

Existing System

Denial-of-service comes in various forms and services. There are three types of attack with different motives are as follows:

- **Consumption of Scarce , limited or non-renewable resources:** Some of things such as network bandwidth, memory, disk space, CPU time, data structures, access to computers and networks and other environmental resources.
- **Network Connectivity** DoS attack mainly takes place in network connection. Example of this type of attack is ”SYN Flood”.

  Attackers machine has established connection with the victim machine such a way that connection is half open. Victim machine reserves limited number of data structures which requires to complete the connection. This attack results in denying of legitimate connections leaving half-open connections. Kernel level data structure is been consumed by the intruder.
- **Using your own resources against you** Generation of excessive mail messages.Placing files in anonymous ftp areas or network shares.Intentionally generating errors that are already logged.
- **Bandwidth Consumption** A large number of packets of ICMP ECHO is been directed to the network resulting into the consumption of network bandwidth.

- **Network Connectivity** Intruders may consume some other resources that are necessary for system operation. Example, Limited data structures are available to hold the process information such as identifiers, entries and process slots. It may be just created by writing the script that copies itself. This can be sometimes prevented by the quota facilities provided by the operating system. If the table is not filled by copying the scripts then CPU may consume large number of processes and associated time between switching.

  Even disk space is consumed in numerous ways:

  - Generation of excessive mail messages.
  - files are placed at anonymous ftp areas.
  - Intentionally generating errors that are already logged.

  If there is no bound on amount of data written on disk it can lead to denial of service. This may even cause system crash by sending malicious data over network.

  The attack is likely to take place once system faces frequent crashes with no specific cause. Some of things that are vulnerable to dos attack or can be used in malicious way are: printers, tape devices, network connections, and other limited resources.

2.1 Issues in existing System

In Packet Marking Techniques, following are the limitations:

- It is more complex and time consuming with less effectiveness.
- It cannot detect attack so fast.

In other traceback Techniques, following are the limitations:

- Packets from legitimate users will be lost.

- Malicious attacker or attacks with lower rate cannot be detected.
- Traceback may be slow or sometimes it may be impossible.

2.2 Categories of DoS Attack

Different categories of DoS attack are as follows [24]:

- Volume Based Attacks: The main aim of this attack is to consume bandwidth of victims website. It includes UDP Floods , ICMP Floods and so on.
- Protocol Attacks: The main aim of this attack is to consume all the resources of servers such as routers , firewall , load balancers and so on. This includes attacks such as Ping of Death , Smurf , TearDrop and SYN flood attack.
- Application Layer Attacks: Its main aim is to destroy the web application or crash the web services. This type of attack includes zero day attack , vulnerability exploitation and so on.
- Session Exhaustion: It exploits the session limitations by not closing old sessions and opening of new sessions.

2.3 TYPES OF DOS ATTACKS

Different types of dos attacks are:

a. **Application Layer Attack:** The main aim is to flood server with large number of request with resource handling and processing. Examples of such attacks are, HTTP Floods, DNS query flood attacks and slow attacks.

b. **Network Layer Attack:** They mainly aim to exhaust network resources. Examples of such attacks are UDP Flood, SYN Flood, NTP Amplification and DNS amplification attacks. 20 to 40 Gbps traffic events are enough to shut down network resources.

c. **Buffer overflow Attack:** The attacker would exploit the vulnerability by sending the large amount of data it can handle. Some of its characteristics are: Sending large number of ICMP messages, sending emails with 256 characters to netscape and Microsoft mail messages.

d. **Smurf Attack:** In this type of attack, a large number of ICMP packets are broadcasted to a computer network with victims spoofed source IP address. With this flooding, spoofed host will not be able to distinguish or receive real traffic.

   some of the mitigation technique for smurf attack is [20] :

   - to block directed traffic entering into the network.
   - configure host and routers in such a way that not to respond to ICMP echo packets.

e. **SYN Attack:** A limited buffer space exists for the rapid hand shaking of messages by setting up sessions. This packets consists of sequence number for exchange of messages. A large number of packets are send and then not responded leaving large number of packets in buffer not permitting the legitimate requests.

f. **Teardrop Attack:** The IP Protocol packets are divided into fragments. This packets are identified by the offset at the beginning of packets. The attacker puts an confusing value to the second or later fragments, which leads to system crash.

g. **Viruses** The viruses replicate across a network in various ways where a host is targeted. In such attack depends on severity, attacks can be hardly noticed.

h. **Physical Infrastructure attacks:** Snipping of fibre optic cable is included in it. Such attacks can be mitigated by rerouting the traffic.

2.4 Performance Metrics of Various types of DoS attack

a. **ICMP Flood Attack**

Ping and its variation hping command are used to check the services of any particular system. Any packet can be maximum of 65,535 bytes. Communication between systems can be carried using ICMP Ping request and ICMP Ping reply. Attacker floods the system for sending thousands of packets to server using spoofed IP Address. So that replies are send to spoofed address. By using ping command to flood with excess number of packets the resources are consumed. Snort rule for detecting ICMP Attack is:

Alert icmp any $\rightarrow$ 192.168.1.0/24$any(msg : ICMPattackdetected;$
$sid : 10000001; rev : 001;)$

Performance Matrix for ICMP Flood Attack

Packet Size(bytes)	CPU Utilization(%)	RAM Usage(Mb)	Network Usage(Mb/S)
10	60	213.5	6
100	68	213.8	10
500	79	450	18
1000	82	520	25

b. **Smurf Attack**

The ICMP echo request is broadcasted with victims IP Address. All the Intermediate machines respond with ICMP echo reply. This leads to flooding of network with thousands of reply. By spoofing the source IP Address same as destination IP Address the resources are exhausted.

Snort rule for detecting Smurf Attack is:

Alert icmp any any $\rightarrow$ 192.168.1.0/24$any(msg : Smurfattackdetected;$
$itype : 8; Sid : 5000002; rev : 1;)$

Packet Size(bytes)	CPU Utilization(%)	RAM Usage(Mb)	Network Usage(Mb/S)
10	15	680	1
100	36.6	710	1.8
500	48	724	2.8
1000	65	742.8	3.3

Performance Matrix for Smurf Attack

c. **HTTP DoS**

HTTP Flooding is been created by use of Zombies i.e. Ufonet. Valid or Invalid Http request are sent to server by using three way handshake communication. By using zombie such as ufonet to perform HTTP DoS attack on Server by generating valid or invalid HTTP Request.

The following rule detects a pattern GET in the data part of all TCP packets that are leaving 192.168.1.0 network and going to an address that is not part of that network. The GET keyword is used in many HTTP related attacks; however, this rule is only using it to help you understand how the content keyword works.

Alert tcp 192.168.1.0/24 any $\rightarrow$![192.168.1.0/24]$any(content:$"GET";
$msg:$"$GETmatched$";)
$Thefollowingruledoesthesamethingbutthepatternislistedinhexadecimal.$
$alerttcp$192.168.1.0/24any $\rightarrow$![192.168.1.0/24]$any(content:$"|474554|";
$msg:$"$GETmatched$";)

Performance Matrix for Http DoS Attack

Packet Size(bytes)	CPU Utilization(%)	RAM Usage(Mb)	Network Usage(Mb/S)
10	30	600	2.1
100	45	630	2.8
500	65	685	3.5
1000	75	700	4

d. **TCP SYN Attack**

The Basic step of three way handshaking is exploited. For communication purpose

between servers TCP SYN and TCP ACK messages are exchanged. Attacker spoofs the IP Address so the SYN ACK packets are send to victims (spoofed) Address which completely fill ups maximum limit of SYN ACK Packets. Since packets waits for ACK until it times out and get dropped, victims machine is flooded with illegitimate request and would not be able to serve legitimate request.By exploiting the basic three way handshake the attack has been performed and has been monitored using ganglia.
Snort rule for detecting TCP SYN Attack is :

Alert tcp any any $\rightarrow anyany(msg : TCPSYNFloodattackdetected; flags : S;$ $threshold : typethreshold, trackby_dst, count20, seconds60; sid : 5000001; rev : 1;)$

Performance Matrix for TCP SYN Attack

Packet Size(bytes)	CPU Utilization(%)	RAM Usage(Mb)	Network Usage(Mb/S)
10	55	700	5
100	70	680	5.2
500	79	747	4.9
1000	85	790.5	5.78

e. **UDP SYN Attack**

Since UDP is connectionless protocol, the attacker generates enough UDP Packets to a random port in Victims Server. On the Victim Side, it will check for application that will be waiting for that particular packet unless it realize there is no application waiting for it. So ICMP with destination unreachable is generated to source address. If enough number of Packets are received at the victim end, the system would be flood and would be down.The ports that are open in victim side is targeted. Enough UDP Packets that can flood the victims server are generated which would exhaust all the available resources such as CPU, bandwidth and memory.
Snort rule for detecting UDP SYN Attack is:
Alert udp any any $\rightarrow 192.168.1.0/24any(msg : Landattackdetected; flags : S; threshold :$ $typethreshold, trackby_dst, count20, seconds60; sid : 5000003; rev : 1;)$

Performance Matrix for UDP SYN Attack

Packet Size(bytes)	CPU Utilization(%)	RAM Usage(Mb)	Network Usage(Mb/S)
10	60	420	2.5
100	65	432	4
500	75	450	6.5
1000	62	480	7.9

f. **Land Attack**

The source IP is spoofed as of Destination IP. So the machine send huge request to itself and this conflict cannot be resolved at last victim gets crashed or rebooted.Spoofed IP Address that is same as Victim is used by attacker so that request is send to itself and all the resources gets consumed.

Snort rule for detecting Land Attack is:

Alert tcp any any $\rightarrow anyany(msg : Landattackdetected; flags : S; sameip; sid : 5000000; rev : 1;)$

Performance Matrix for Land Attack

Packet Size(bytes)	CPU Utilization(%)	RAM Usage(Mb)	Network Usage(Mb/S)
10	65	350	2.9
100	71	368	3.1
500	73	371	3.5
1000	75	380	3.7

2.5 DoS attack detection algorithm

There are several type of DoS attack as mentioned above. Based on the motives and techniques used severity varies. Different techniques are used to detect the severities of this variations of attack.

2.5.1 Fuzzy Algorithm

In this algorithm , trapezoidal shape is used including four parameters a , b, c and d. This algorithm is used to calculate the probability of attacks [23].

if $(data > a)$ && $(data < b\ )$ **then**

$prob$ =data - a/(b-a);

else

  if $data \geq b$ and $data \leq c$ **then**

    $prob = 1$;

  end if

else

  if $data \geq c$ and $data \leq d$ **then**

  $prob$ =d - $data$/(d-c);

  end if

else

  $prob = 0$;

end if

2.5.2 IBRL Algorithm

This proposed model monitoring system collects real time traces of traffic to monitor.

According to the IBRL algorithm, the throughput on the Serial Interface1 of edge router (S1/0) is checked such that if it is greater than both the throughput of Serial Interface2 (S1/1) and Serial Interface3 (S1/2) of the edge router, the link utilization of the Serial Interface1 (S1/0) is also checked. If the link utilization exceeds 95% of the bandwidth capacity the rate limit rules are applied in Serial Interface1 (S1/0). This procedure is repeated for all other interfaces simultaneously[22].

The severities of dos attack varies and with it some of the scripts can even harm or freeze the system or can bring down the server.

Since it is found that TCP SYN Flood is more severe than other attacks. There

```
if (P_th (S_1/0) > P_th (S_1/1) and P_th (S_1/0) > P_th (S_1/2))
    check B (S_1/0)
    if (B (S_1/0) > b) then
        Rl -> S_1/0
    endif
else
if (P_th (S_1/1) > P_th (S_1/0) and P_th (S_1/1) > P_th (S_1/2))
    check B (S_1/1)
    if (B (S_1/1) > b) then
        Rl -> S_1/1
    endif
else
if (P_th (S_1/2) > P_th (S_1/0) and P_th (S_1/2) > P_th (S_1/1))
    check B (S_1/2)
    if (B (S_1/2) > b) then
        R_1 -> S_1/2
    endif
endif
```

Figure 2.1: IBRL Algorithm [22].

are various Algorithm to detect TCP SYN Flood attack.

2.6 Intrusion Detection System in cloud

2.6.1 Intrusion Detection System

Introduction

As cloud has been vulnerable to security attacks due to its open and distributed nature , a strong defense mechanism is always required to protect from such mechanisms. Use of firewall or antivirus single handedly will not eradicate all the vulnerabilities. A strong defense is required to handle such security loop holes for that intrusion detection system can be deployed at different network locations based on its requirement.

Intrusion detection system is a hardware or a software application used to monitor and detect malicious behaviour of traffic.Every year CERT Reports increasing amount of attacks.

Past research

Multithreaded IDS have been proposed for the distributed system. It is mainly used to detect masquerade attacks, host and network based attacks [18].

Jabez suggested an approach to use the outlier detection for the network intrusion detection system [9]. The paper focused on little variation of attacks ,low false alarm rates.

Narwane proposed knowledge and behavior based approach to detect anomalies. Behavior of system is observed and slight change in behavior will trigger the alarm and if changed behavior remain unnoticed then network packet is been compared with database of vulnerabilities which will raise an alert [17].

Bamakan demonstrated two methods name multiple critical linear programming and swarm particle optimization to improve performance by decreasing false alarm rate [10].

An another approach is network based signature which is palced at each node to detect SIP Flooding attack [20]. Modi has even proposed hybrid technique to detect major attacks and should be located at server [3].

Hybrid technique of two approach covariance matrix based and entropy based system has been proposed [22].

Chia-Mei, Guan, Yu-Zhi, and Ya-Hui (2012) investigated the problem of sequence of in Cloud. An attacker can maliciously combine multiple security vulnerabilities and by adapting persistent attack approach of sequence network may harm the cloud.Thus, they proposed Hidden Markov model to detect such attacks by examining the attack plan at different stages and analyzed logs to identify attack sequences.

S.H.C Haris et al. suggest that IP Header and TCP header payload are used for detection of TCP SYN Flood. Port, flag, IP address, Protocol behavior and so on are some of the key features used for attack detection. The focus of this paper is limited to detect attack in the local area network in File Transfer Protocol and has lower detection rate. The packet captured using tcpdump are filtered using packet filtering algorithm and thus would raise alarm based on deviation from normal behavior[25].

Y. Ohsita et al. suggest to consider arrival time variations. This proposal is limited to detect normal TCP SYN Packets as lower rate traffic cannot be detected as it follows normal distribution model. By normal model distribution, the mechanism can detect attack accurately [26].

H. Wang et.al suggest that the detection system should be kept at the edge of routers or firewalls or proxy at the front end. It analyze the TCP SYN FIN pairs and the change in the sequences. Various alerts are generated based on the events and source of flooding can be identified. Thus the limitation of it is that system is more prone to flooding attacks but it does remove the overhead. Along with detecting attacks by generating alarms even the source location can be found using this technique[27].

M.Durairaj et al. proposed ThreV algorithm for detecting MAC spoof DoS attack as MAC address can easily be spoofed. The paper focuses on existing Infrastructure. Hybrid Mechanism is proposed which is amalgamation of four algorithms such as ThreV, Alternative Numbering Mechanism, Traffic Pattern Filtering and Letter envelop protocol. The Basic Identity Check tables is compared with MAC address of all users in WLAN and based on that Intruders can be checked. The benefit of this technique is that it is deployed with minimum packet loss, reduced control overhead with reduced in packet drop and delay[28].

Maciej Korczynski et.al suggest that scheme that relies on sampling rate. To validate connection, TCP Packets are examined to check for ACK Segments coming from server. This method is effective when the rate of incoming packets is been controlled and then further compared with other detection methods. The ACK flag is mainly examined with set on means that connection is legitimate. Although this method is very effective by decreasing false positive rate but some information is lost while sampling data[29].

D.M.Divakaran et.al suggest to use exponential back off property of TCP segments to determine high intensity of attack. Linear Predictive analyze network traffic and various other types of DoS attack. Even the intensity of network can be detected using LP Detection Method. The low and high intensity SYN flooding attacks can be detected. There will be detection delay in source identification of TCP SYN flood[30].

S.H.C Haris et al. suggest that use of payload and unusable area in Hyper Text Transfer Protocol. ToS, IP Header, Unusable area are considered for detecting TCP SYN. To detect the TCP SYN attack it is necessary to recognize normal payload characters else would be time consuming. The need arises to make detection faster and effective[31].

Parasa Harika et al. suggest to count and record SYN packets whose three way handshake is completed. Even all packets that are opposite to SYN packets are recorded. The Proposed Technique is combination of packet filtering and syn flood monitoring [32].

D.D.Rani et al. suggest to check open ports and its active connections in Server. Using Wireshark and IP table rule DoS attack is analyzed. Once DoS attack is detected its prevention can be done using shell scripts to block such network traffic. The experiment for detection is limited to client server program [33].

D.S.Rana et al used Wireshark to detect TCP SYN flood attack. The attack has been generated by Shell Script using random number function so that the request comes from Random IP address. Use of Inbuilt functions in Linux such as netstat is done. Around 2000 to 7000 packets are captured at network interface [34].

V.A.Siris et.al evaluated adaptive threshold algorithm and cumulative sum algorithm for change point detection. Adaptive threshold algorithm checks for network traffic and compares SYN packets with the threshold value. When the number of SYN Packets exceed number of FIN Packets the change has been noted using cusum algorithm. For low intensity attack there is degradation in performance. It is efficient for detecting high intensity attacks without being more complex [35].

V.L.L.Thing et.al proposed use for bloomed filter. The outgoing SYN packets values must be equal to incoming SYN ACK values. The technique is more reliable in detecting SYN-SYN ACK detection mechanism rather than SYN-FIN/RST detection mechanism. SYN-FIN/RST fails to detect Bot Buddy attack[36].

2.6.2 Intrusion Detection Techniques

Different techniques used for detection are as follows:

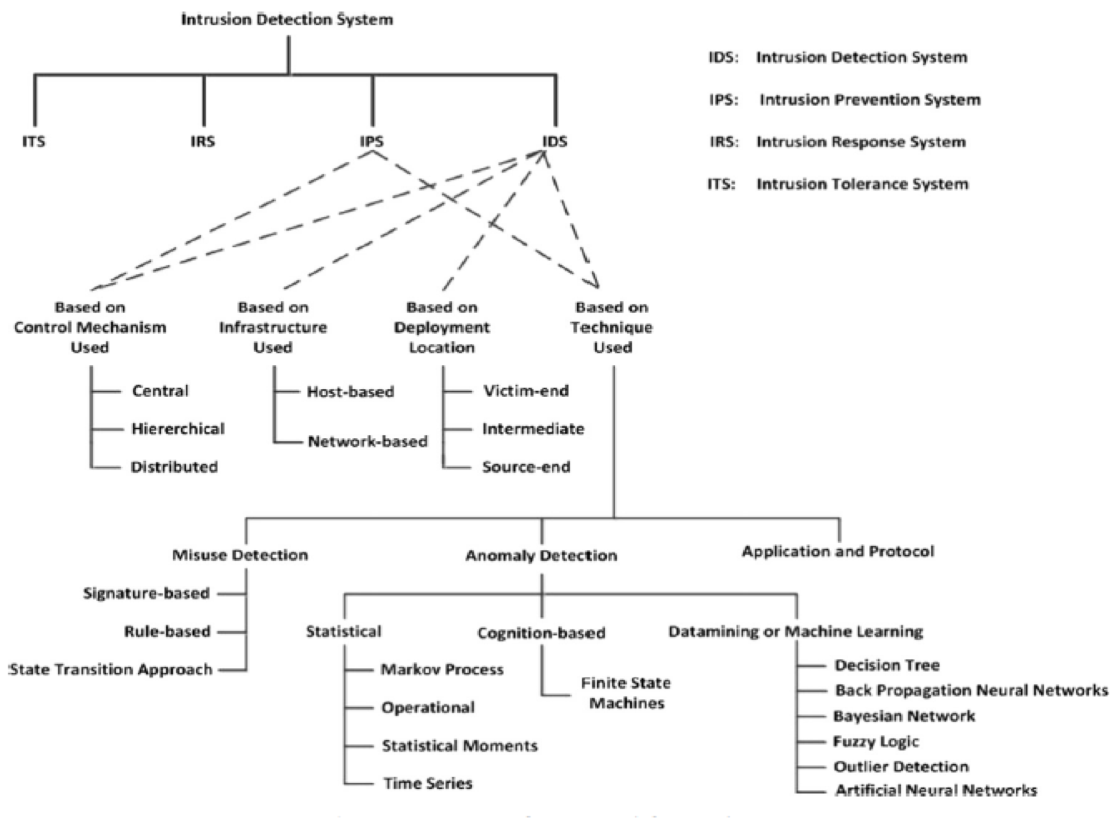

Figure 2.2: Taxonomy of IDS Techniques [6]

This above techniques are combined or used individually for detection of known as well as unknown attacks. This techniques may even require to train the network from past attacks. IDS and IPS have always behaved as support system for the network. However , they may not be even completely reliable. So some additional features are always required to secure the network.

2.6.3 Limitation of IDS

Limitations of IDS are as follows:

- They are not scalable for large or distributed network.
- Traffic audit data changes with time interval making it difficult differentiate normal traffic from anomaly[5].
- They cannot hide security vulnerabilities in network protocols.
- Its significantly error prone i.e. more number of false positive[17].
- Monitoring user behavior is difficult [17].
- Human intervention is always required.
- Timer increases double check points are kept so it degrades the performance[22].
- Difficult to detect unknown and novel attacks, requires huge execution time and is less accurate[9].
- It cannot be complete reliable solution against security threats.
- Encrypted packets are not processed which can cause intrusions in network.
- Complexity increases as both techniques are combined[3].

Chapter 3

Literature survey table

Paper title	Abstract	Technicalities	Conclusion	Future work
Security in Cloud Computing: opportunities and challenges 2015 [6]	Cloud computing models, types of cloud and its characteristics. Issues at Communicational (DoS,man-in-middle ,masquerade , eavesdropping , cryptographic algorithms), architectural(virtual network ,security misconfigurations, virtualization issues,VM image sharing, Migration) and contractual and legal issues	Data visibility and other operations due to multitenancy. On demand self service may lead to unauthorized access to management interface,operational dependency among clouds.	Threats should be well understood and deal accordingly	Communication level challenges can be eliminated. Different techniques can be applied along with cryptographic algorithms and key management.
Understanding DDoS attack and its effect on cloud computing[14]	Unable all the resources, finding of bugs or issues in software implementation, deplete all bandwidth or resources	Different types of DoS attacks: Bandwidth Depletion attack ,flood attacks,amplification attacks,Resource Depletion Attack ,IP address attack	Prevention System: Ingress and egress filtering,, Route based distributed packet filtering,Secure Overlay Service. Detection Techniques: Anomaly detection,NOMAD, packet sampling and filtering with congestion , DWARD , MULTOPS,misuse detection	NA.

IDS: Anomaly detection using Outlier detection approach 2015	For encountering larger data sets , statistical approaches and rule based expert systems were not accurate. Expert Systems based on rules will detect the known intrusion in high rate and will not identify intrusion. Monitor the current status of user	Detect masquerade attacks ,malicious use ,leakage ,service denial , unauthorized users break in, identifies zombies based on server connection,genetic algorithm uses fitness function for estimating the rules.Based on Fuzzy clustering and ANN approach.	Proposed System identifies all types of attacks such as probe , DoS,U2R and R2L. depends on outlier values.IDS performance can be improved.	Distance computation function used between trained model and testing data.
A New Intrusion Detection Approach using PSO based Multiple criteria Linear Programming[15]	Multiple criteria linear programming is a classification method based on mathematical programming which has been showed a potential ability to solve real-life data mining problems,High classification accuracy and low false alarm rate are two main characteristics.	Computational Intelligence methods such artificial immune systems,artificial neural network,swarm intelligence and soft computing showed better performance.	Better performance based on accuracy and running time was examined by KDD cup	Multi class classification can be applied on KDD cup 99 to examine performance on different attacks.

Stealthy Denial of Service strategy in cloud computing [16]	Effects pay by use module , service degradation is also considered as vulnerability, Malicious activities are in a stealthy fashion to elude security mechanism by orchestring and timing attack	Vulnerability metric for maximum performance degradation .evaluating vulnerability of open and closed hash	Implementing a stealthy behaviour by slowly increasing polymorphic behaviour, exploiting cloud facility	Detect attacks at application level. detect spidas attack in cloud.
Comparison of Network Intrusion detection system in cloud computing environment [23]	Famous NIDS such as Tcpdump, snort, Network Flight Recorder are contrasted. It identifies packets based on the signature.	Use of network monitoring tools such as Snort,Network Flight Recorder and TCPDUMP.	Comparison of network monitoring tools is done	NA.
CIDS a Framework for intrusion detection in cloud system [24]	Network based IDS based on signature and target. Either Knowledge base or behavior base technique is used to detect attack.	Description of components of alerts summarizer to cloud administrator	Use of CIDS to monitor the system for detecting the intrusions	Apply three proposed detection model by making it more secure.

Intrusion detection in cloud [18]	Some known IDS are Snort, F-Secure Linux Security, Samhain. Vulnerable to cross site scripting, buffer overflow, gaining access to hardware layer thus compromising VM . Remote control is combination of VM control , monitoring and configuration. Use of User-Mode-Linux for implementation of features.IDS faces challenges:1) no standardized o/p. 2) communication and deployment scheme should be flexible. Sorting, filtering and tagging are some of the approaches event correlation.	architecture of deploying intrusion detection system in cloud	IDS deployed at each layer of cloud to gather alerts from each layer and correlate them	Correlation of alerts such as filtering ,sorting and tagging at each layers and deploying IDS through different techniques
A feature selection algorithm to intrusion detection based on cloud model and multi objective particle swarm optimization [19]	Use of EFSA-CP algorithm, best fitness value is used for convergence. Algorithm extracts important features which leads to increase in speed and safety analysis.	use of EFSA-CP algorithm	Use of feature selection algorithm to accurately determine the dimensions of data	Comparing performance of detection on different types of attacks.

CIDD : A cloud Intrusion detection dataset for cloud computing and masquerade attacks [20]	Knowledge and behavior based audit data collected from unix and windows users. Log analyzer and correlator system to extract and correlate user data.Some basic information like login time , session duration and commands are issued. Events that are covered under logged falls in: account management , process tracking ,logon and system events,object access. Different data set are used to detect masquerade attacks: SEA,Greenberg,purdue,RUU. LACS Parses binary log files.	Use of Data set to collect knowledge and behavior based from windows and unix based user.	Analyzing of binary log files and correlating them	Developing the correlator and log analyzer to parse and analyze log files and network packets.

Table I: literature survey table

Chapter 4

Proposed System

4.1 Working

Initially DoS attack script would be kept running in the network , so that attack can be continously monitored. Script would be kept running in the background to monitor the network , uptime , load and even the used RAM. This will help to detect any other variation of DoS attacks. Even in the background this scripts would be kept running.Web application deployed on server would be the victim .

The request would be containing malicious packet that are sent to the web application. Intrusion Detection System would be deployed before the web application. All the incoming request are inspected at IDS and based on the signature rules they are traced. This packets undergoes through the fuzzy model where the expert rules are predefined and based on the expert rules severity of attack is determined. This would prevent request to enter the web application and would be blocked or delayed based on results.

4.2 Proposed solution

The accuracy of the algorithm is 95% if expert rules are accurate enough to detect the attack severity. It is necessary to focus on the rules as this rules will decide the severity of the attack. The fuzzy model consists of expert rules.

Again the challenge here is the expert rules based on past experience. Rule based detection is done. All possible rules are added to the model so that severity can be defined and detected accordingly.

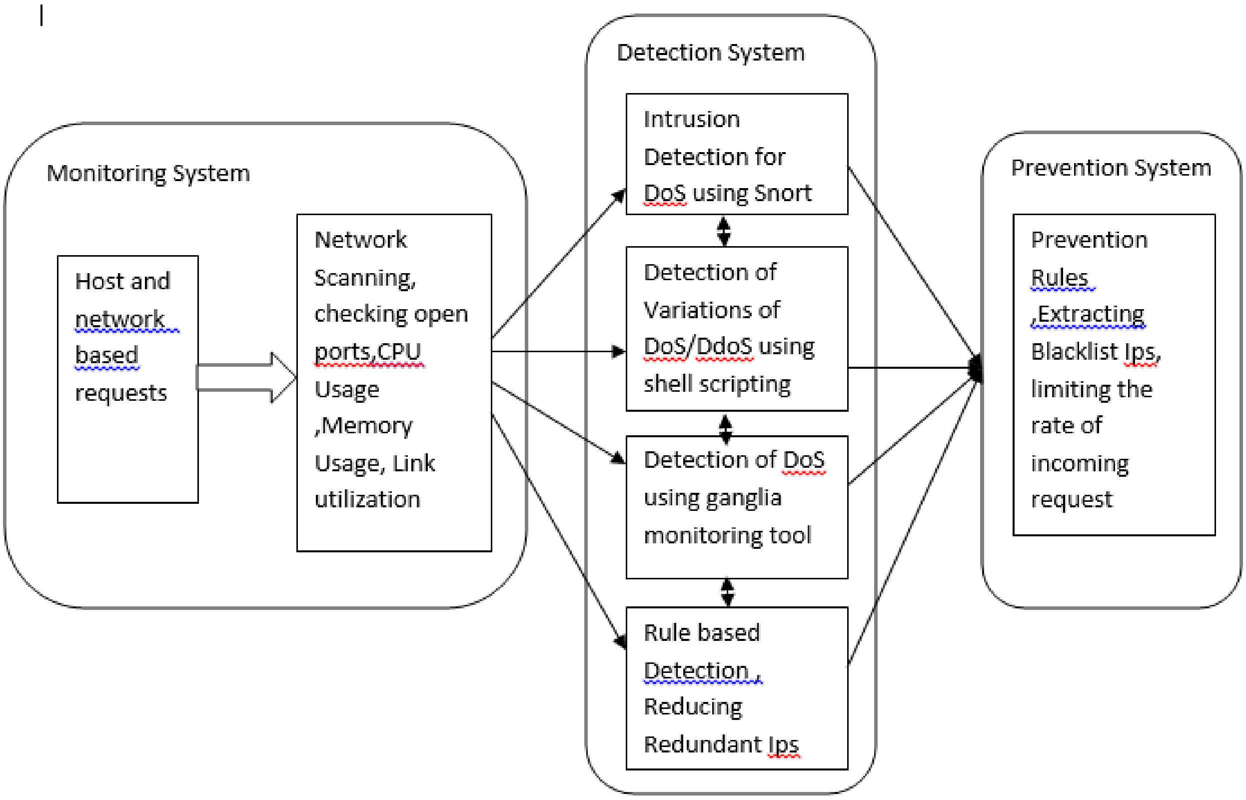

Figure 4.1: Proposed model

4.3 Proposed Flow

All the incoming requests arriving at network are analysed in terms of its CPU Usage , Memory Utilization , Number of Incoming Request , Number of Open Ports and So on. This system's performance has been monitored using ganglia. Further, Various types of DoS attacks has been detected using snort. Events that are generated by Snort are logged into barnyard2. BASE facilitates by providing the graphical detection of network traffic along with identification of unique IP's. Based on Rule based detection , attacker can be identified in network. Redundant IPs, IPs with maximum number of connections and so on are identified. These detected IPs are saved in black list. The attack can be further prevented by limiting incoming request and updating blacklist files.

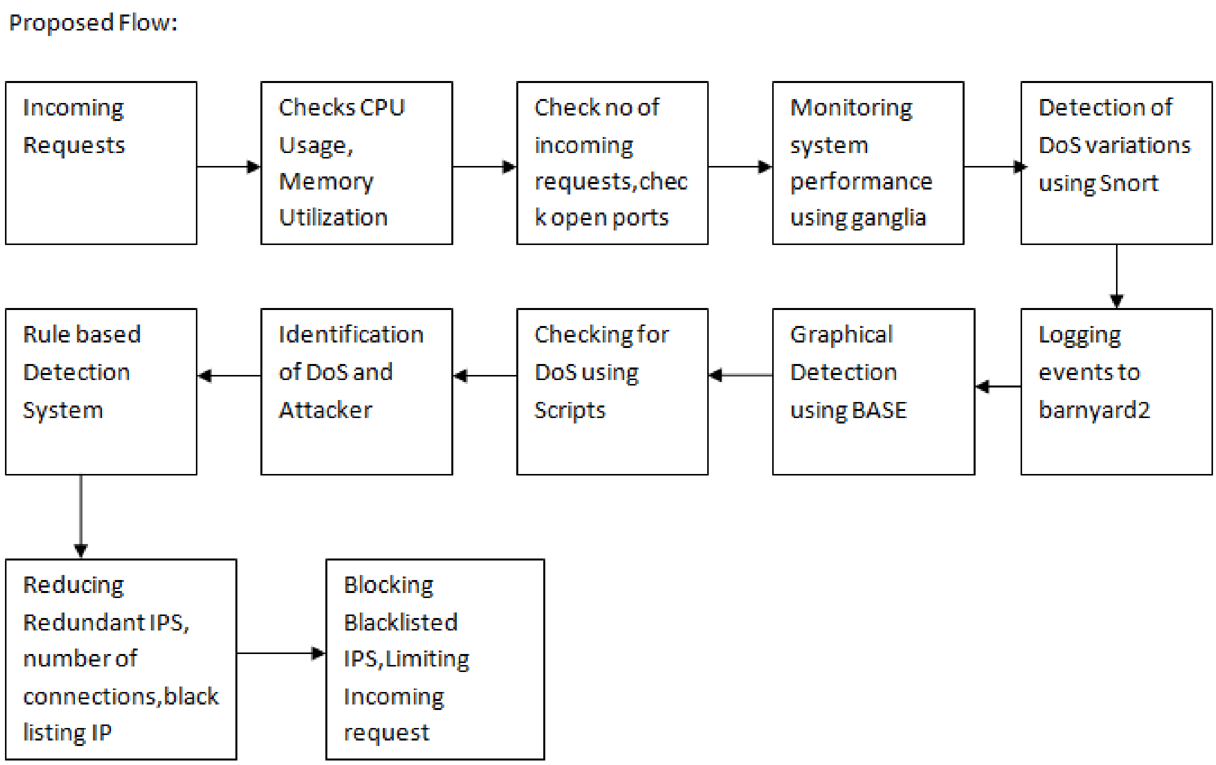

Figure 4.2: Flow Diagram of proposed model

4.4 Implementation

The Experiment has used six host machines deployed in cloud. Host A, B, C, D are attackers with OS Ubuntu 12.04.5 LTS. Host E is IDS which consists of Snort and Rule base detection techniques with OS Kali 2.0 Sana. Host F is client with OS Ubuntu 12.04.5 LTS deployed with ganglia to monitor the performance. Some of the tools used are Snort 2.9.7.6, DAQ 2.0.6, barnyard2 2-1.13, base 1.4.5, LOIC 1.0.8, hyenae 0-1.1, ufonet 0.6, Airmon-ng, Airodump-ng, Driftnet.

The attack is performed on Host F through use of different attack scripts, tools such LOIC, ufonet, hyeane. All the incoming request of Host F is been analyzed and monitored by Host E. Different Rules are configured based on Various Attacks. Both Host E and F are configured with ganglia so that performance can be analyzed.

System implementation is performed with following platform and architecture : Ubuntu 14.04 , kali linux 2.0 operating system is used. =

Name	Version	Purpose
Ubuntu operating system	14.04	
kali linux operating system	2.0	
snort	2.9.7.6	open source NIDS
daq	2.0.6	Makes abstract calls to capture packet libraries
barnyard2	2-1.13	open source interpreter for unified output
base	1.4.5	front end to analyse and query alerts
perl	5.22.2	Programming language
wireshark	1.12.5	Packet Sniffing tool
Tshark	1.12.5	Command line Packet Sniffing tool
TCPDUMP	4	TCP Traffic Intercepting tool
Airodump-ng	4	dump all wireless connection detail
Brupsute	1.6	To Intercept Session of users
Driftnet	-	To Sniff Images from Captured Packets

Table I: Tools used

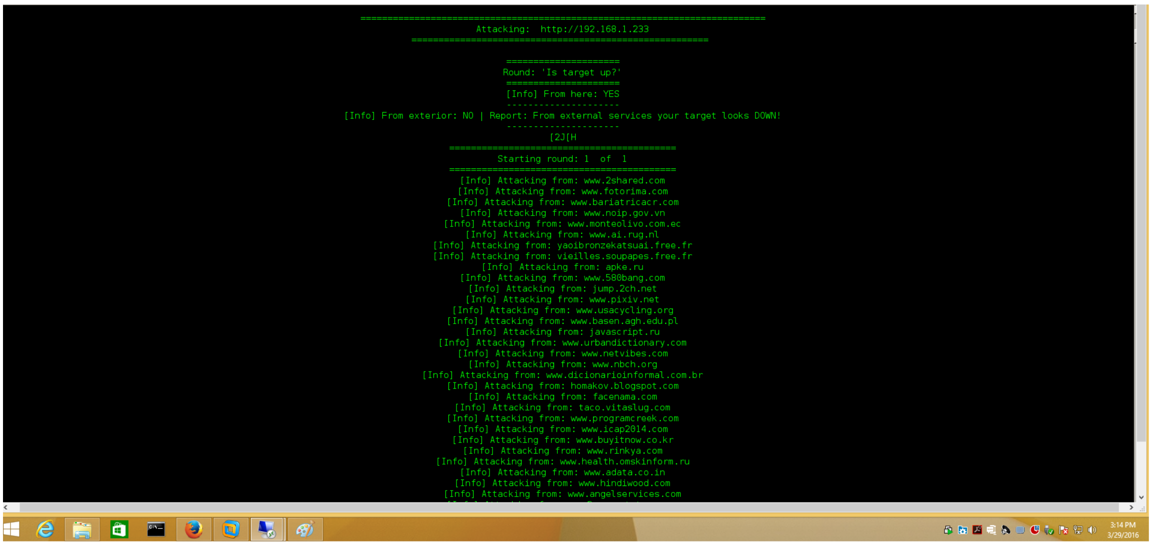

Figure 4.3: Zombie Attack using Ufonet

- Configuring web application on one of ubuntu.
- Scripts are written for Manually and auto monitoring of network.
- scripts for monitoring ping request , server is down , server monitoring log.
- ping of death attack performed along with its detection by snort rules.

4.5 Results Analysis

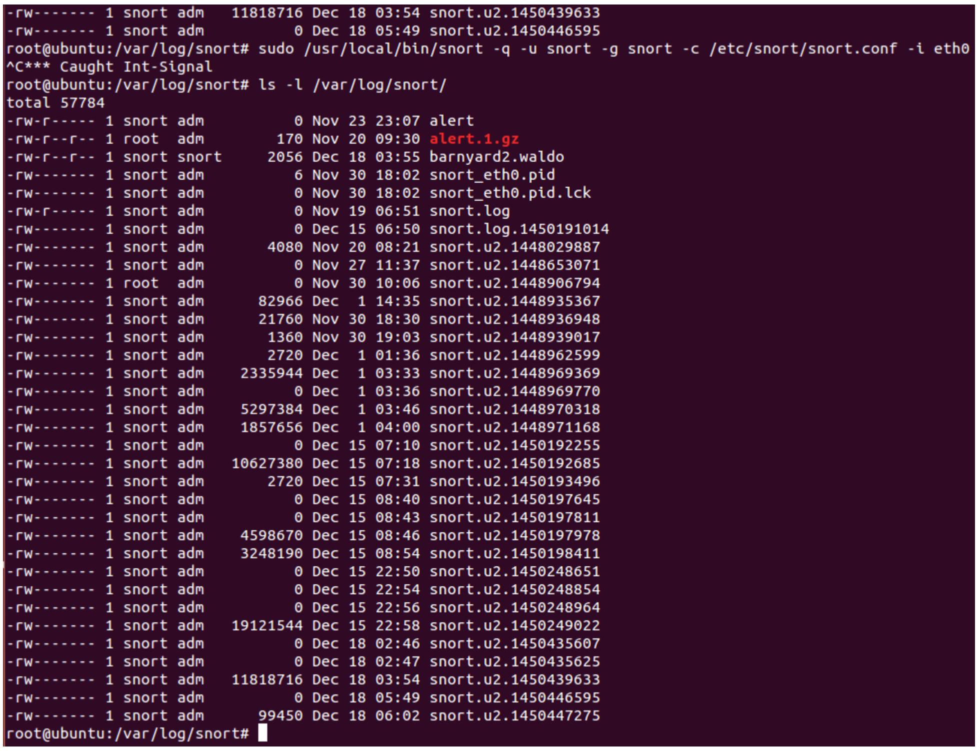

Figure 4.4: Updation of Log files of TCPSYN flood attack in database

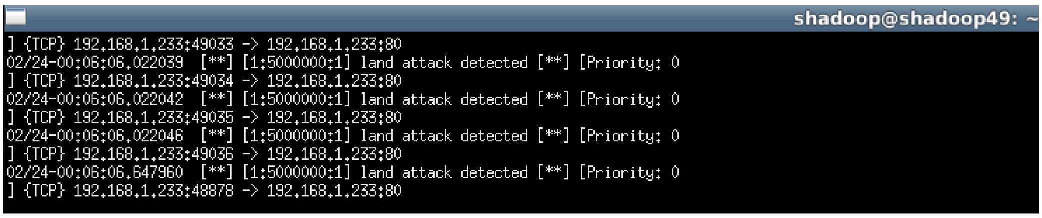

Figure 4.5: Detection of Land attack through Snort

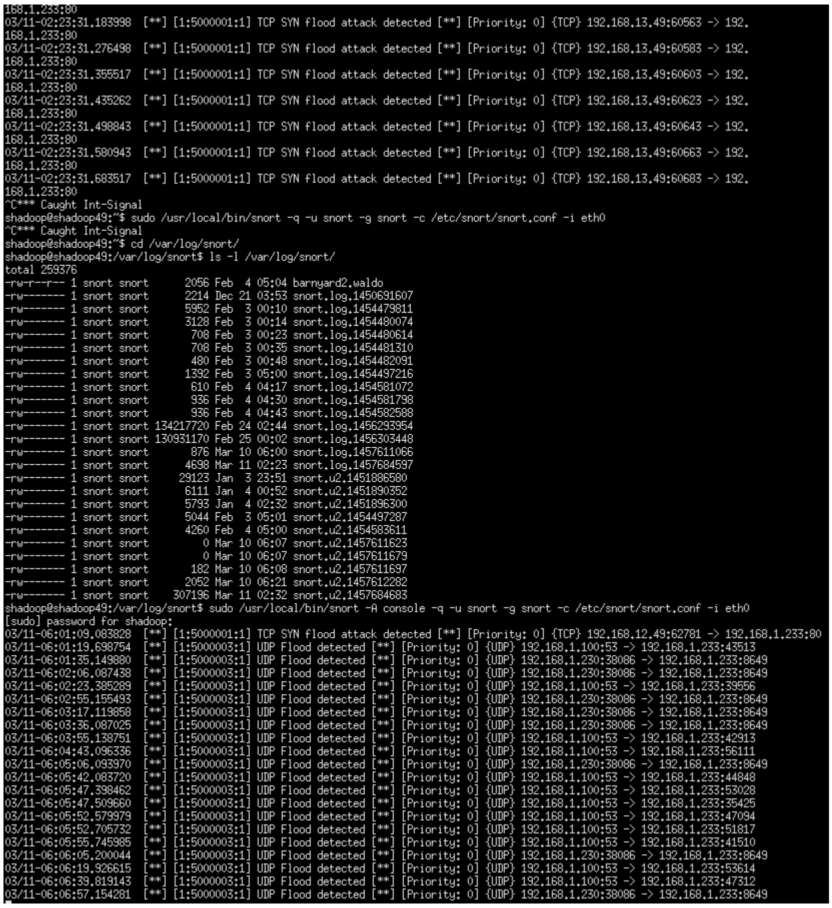

Figure 4.6: Detection of UDP SYN Flood attack

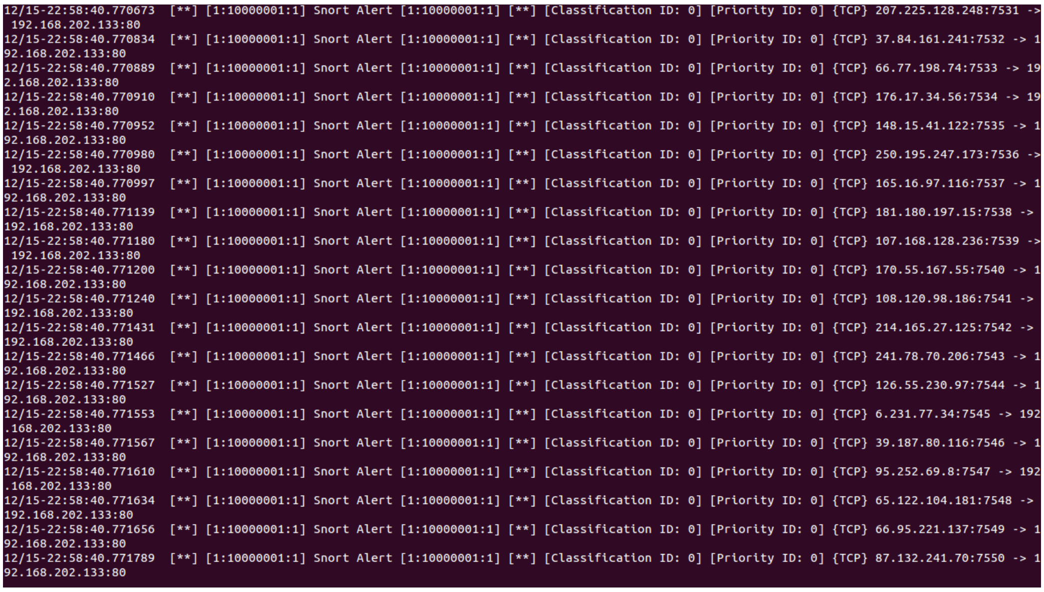

Figure 4.7: Detection of TCPSYN flood attack through Snort

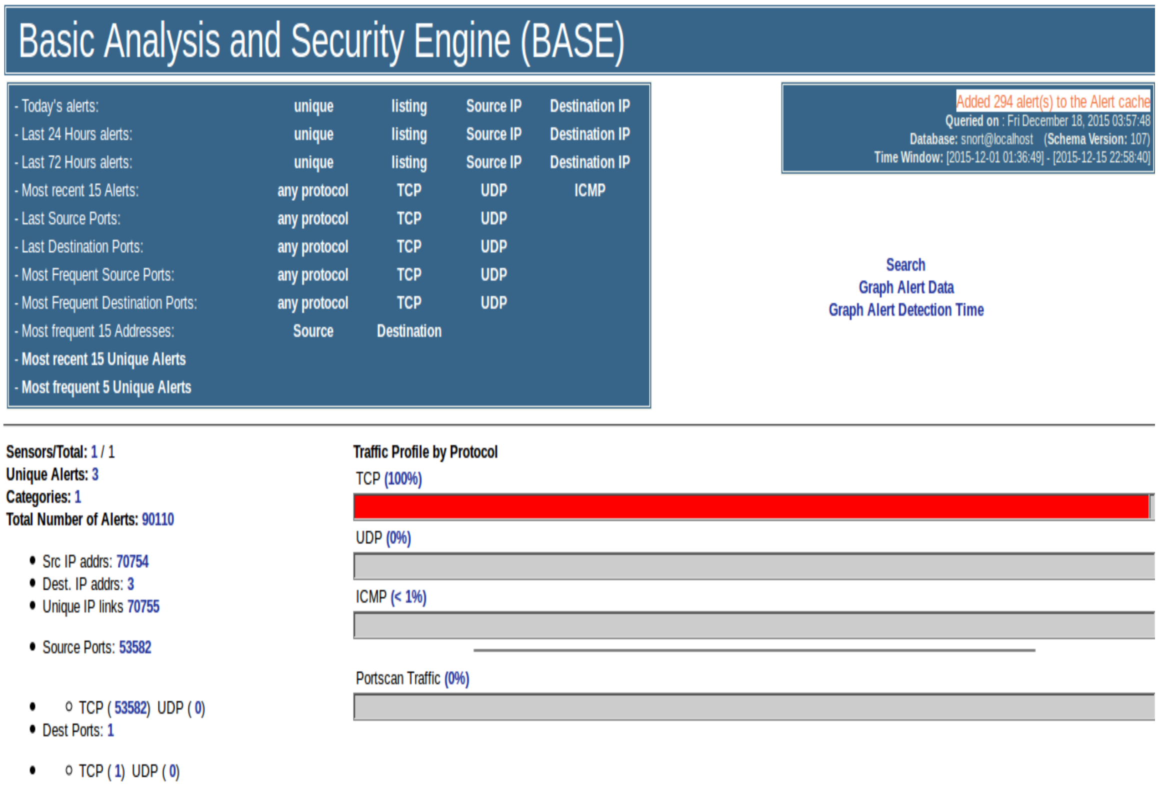

Figure 4.8: Graphical Representation of Detection of TCPSYN flood attack

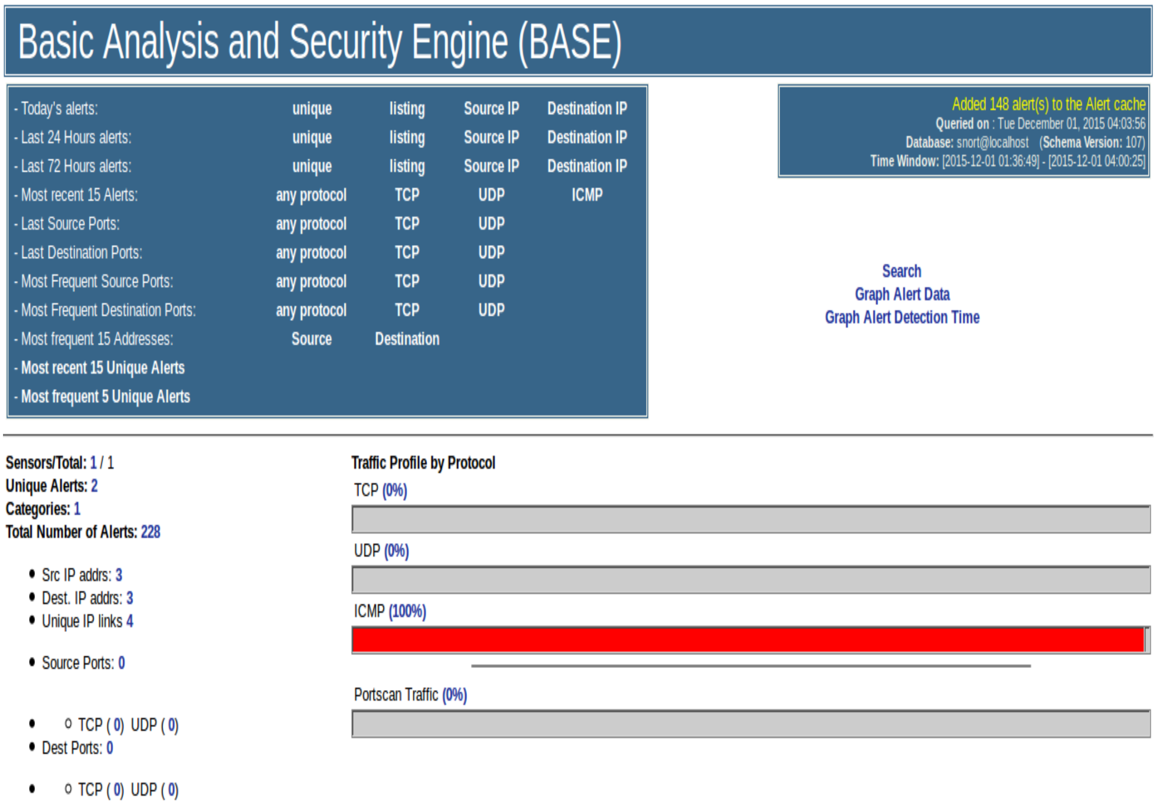

Figure 4.9: Graphical Representation of Detection of ICMPSYN flood attack

11947	14.627158	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11948	14.627984	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11949	14.637266	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11950	14.638983	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11951	14.647360	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11952	14.648181	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11953	14.657475	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11954	14.657984	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11955	14.668754	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11956	14.670728	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11957	14.671589	192.168.1.230	192.168.12.49	T.125	130	T.125 payload
11958	14.678802	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11959	14.679867	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11960	14.688902	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11961	14.689778	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11962	14.698997	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11963	14.699987	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11964	14.709093	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11965	14.709990	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11966	14.719193	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11967	14.719580	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11968	14.722771	192.168.12.49	192.168.1.230	TCP	60	60738 > ms-wbt-server [ACK] Seq=5755 Ack=10796826 Win=1972 Len=0
11969	14.729285	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11970	14.729805	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11971	14.739378	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11972	14.739991	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11973	14.749476	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11974	14.749915	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11975	14.759567	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11976	14.760091	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11977	14.769669	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255
11978	14.770343	192.168.1.241	192.168.1.230	ICMP	1042	Echo (ping) reply id=0x6998, seq=50723/9158, ttl=64
11979	14.779911	192.168.1.230	192.168.1.241	ICMP	1042	Echo (ping) request id=0x6998, seq=50723/9158, ttl=255

Figure 4.10: Detection of ICMP Attack using Wireshark

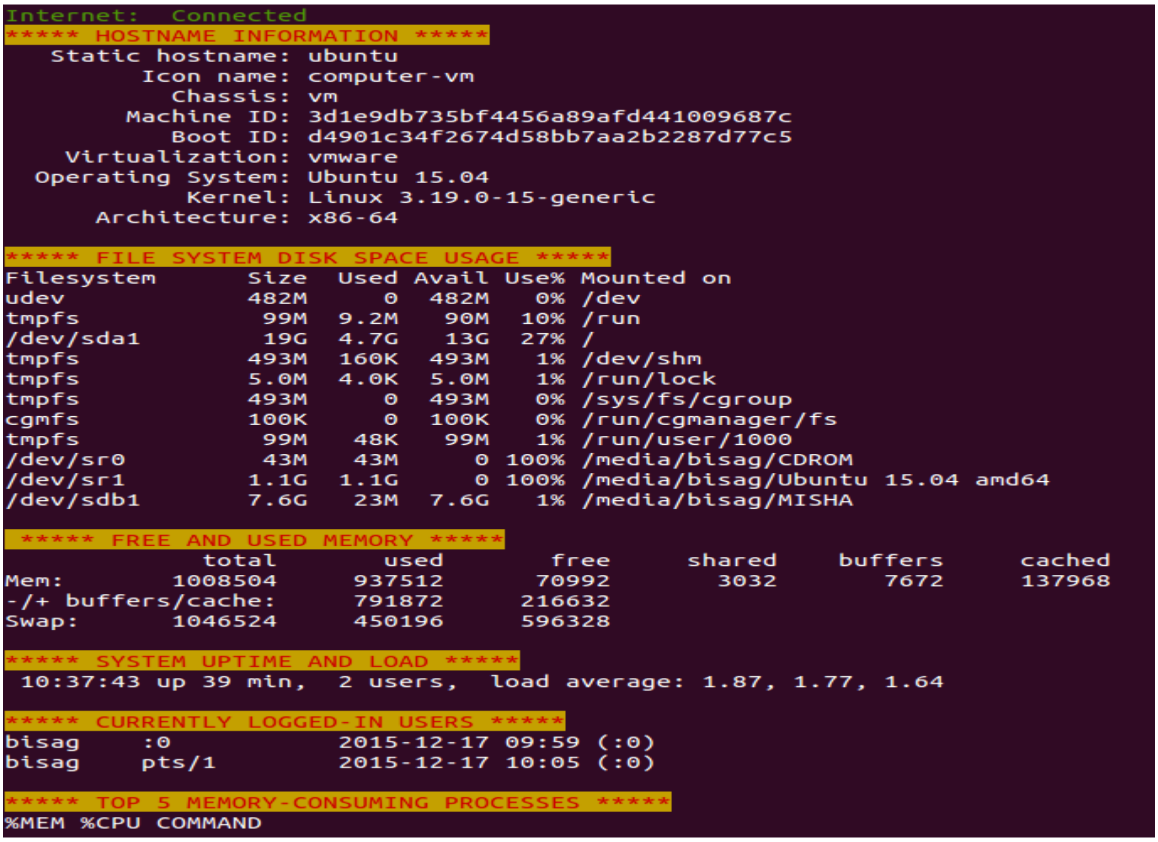

Figure 4.11: System information

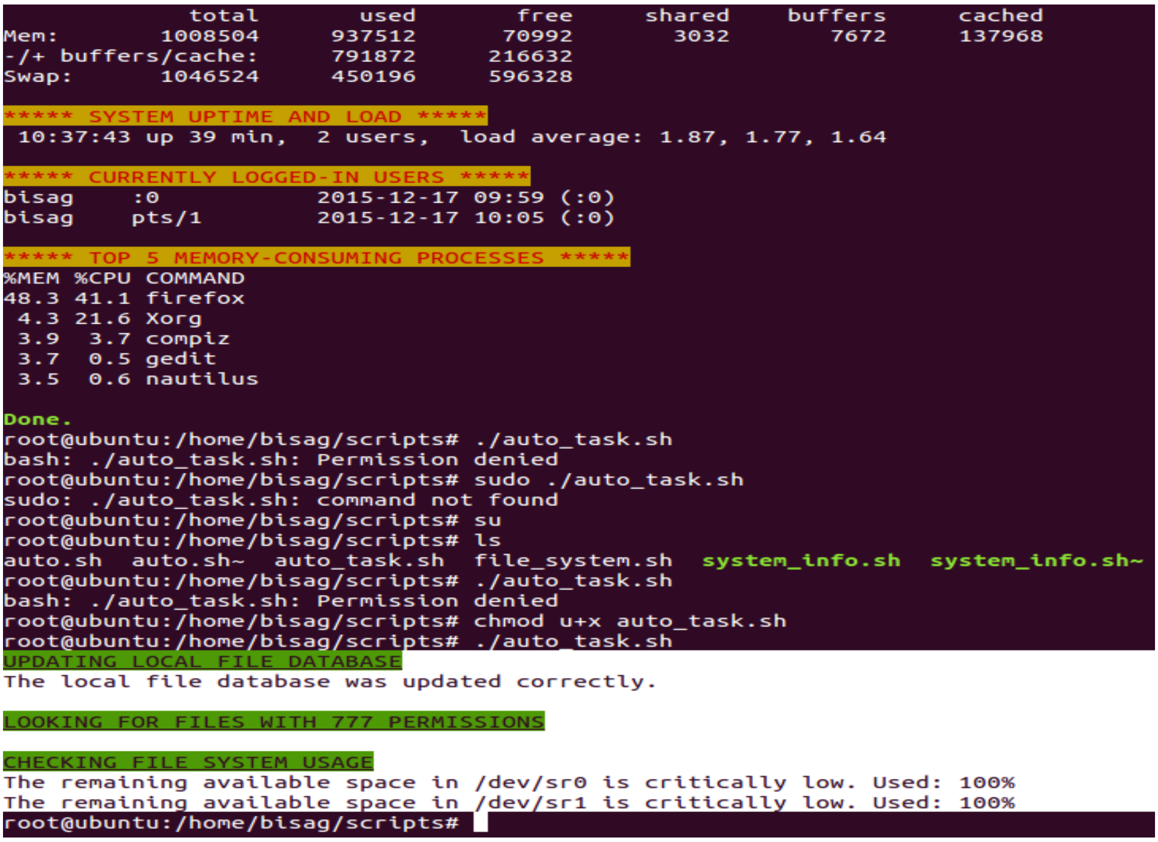

Figure 4.12: Updating local database

Filesystem usage for host **localhost**
Last updated: **Sun Dec 20 06:07:38 PST 2015**

Filesystem	Size	Use %
udev	482M	0%
tmpfs	99M	10%
/dev/sda1	19G	29%
tmpfs	493M	1%
tmpfs	5.0M	1%
tmpfs	493M	0%
cgmfs	100K	0%
tmpfs	99M	1%
/dev/sr0	43M	100%
/dev/sr1	1.1G	100%

Figure 4.13: Alerts generated by snort

```
OS Name : Ubuntu
OS Version : 15.04 (Vivid Vervet)
Architecture :  x86_64
Kernel Release :  3.19.0-15-generic
Hostname :  ubuntu
Internal IP :  192.168.239.134 10.0.3.1
/usr/bin/monitor: line 79: curl: command not found
External IP :
Name Servers :  DO 127.0.1.1 localdomain
Logged In users :
bisag    :0           2015-12-09 22:02 (:0)
bisag    pts/7        2015-12-09 22:03 (:0)
Ram Usages :
             total       used       free     shared    buffers     cached
Mem:          984M       923M        61M       6.9M        20M       229M
Swap Usages :
             total       used       free     shared    buffers     cached
Swap:         1.0G        35M       986M
Disk Usages :
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1        19G  4.6G   14G  26% /
Load Average :  average:0.06,0.16,
System Uptime Days/(HH:MM) :  18 min
bisag@ubuntu:~$
```

Figure 4.14: System Information

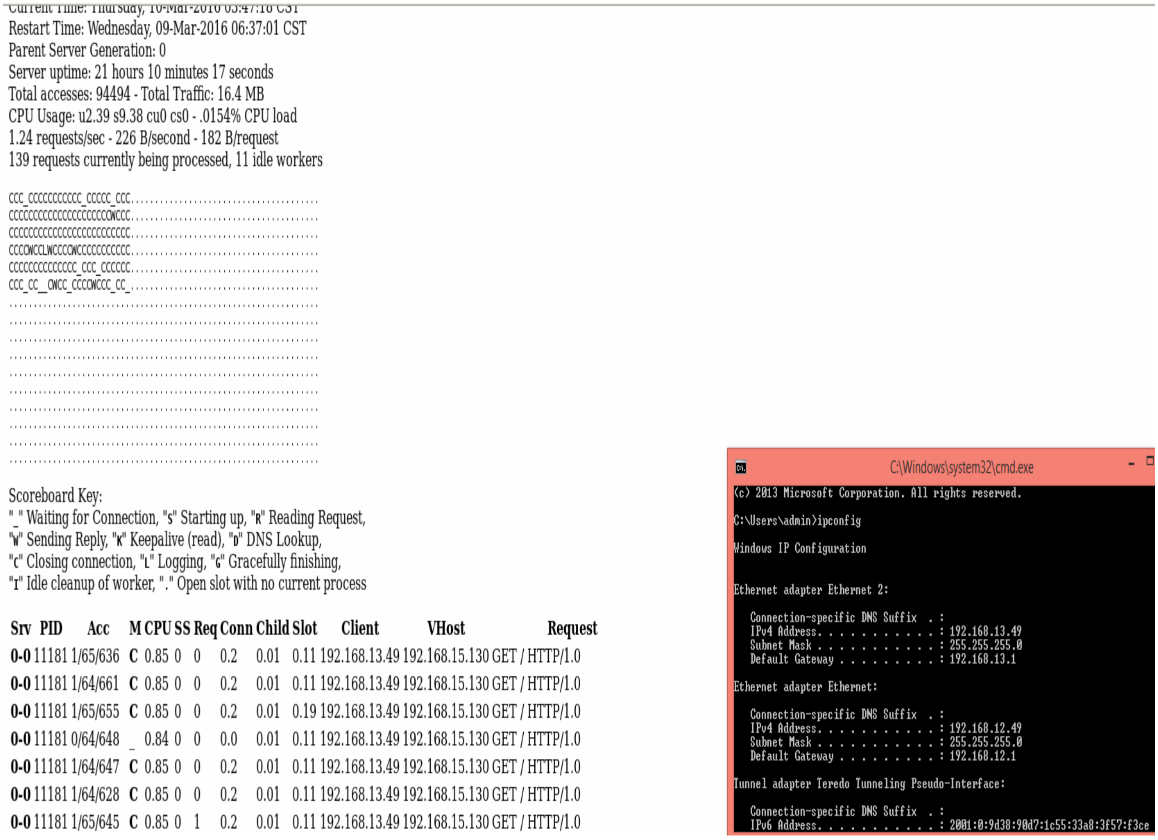

Figure 4.15: TCP SYN Detection Using Apache

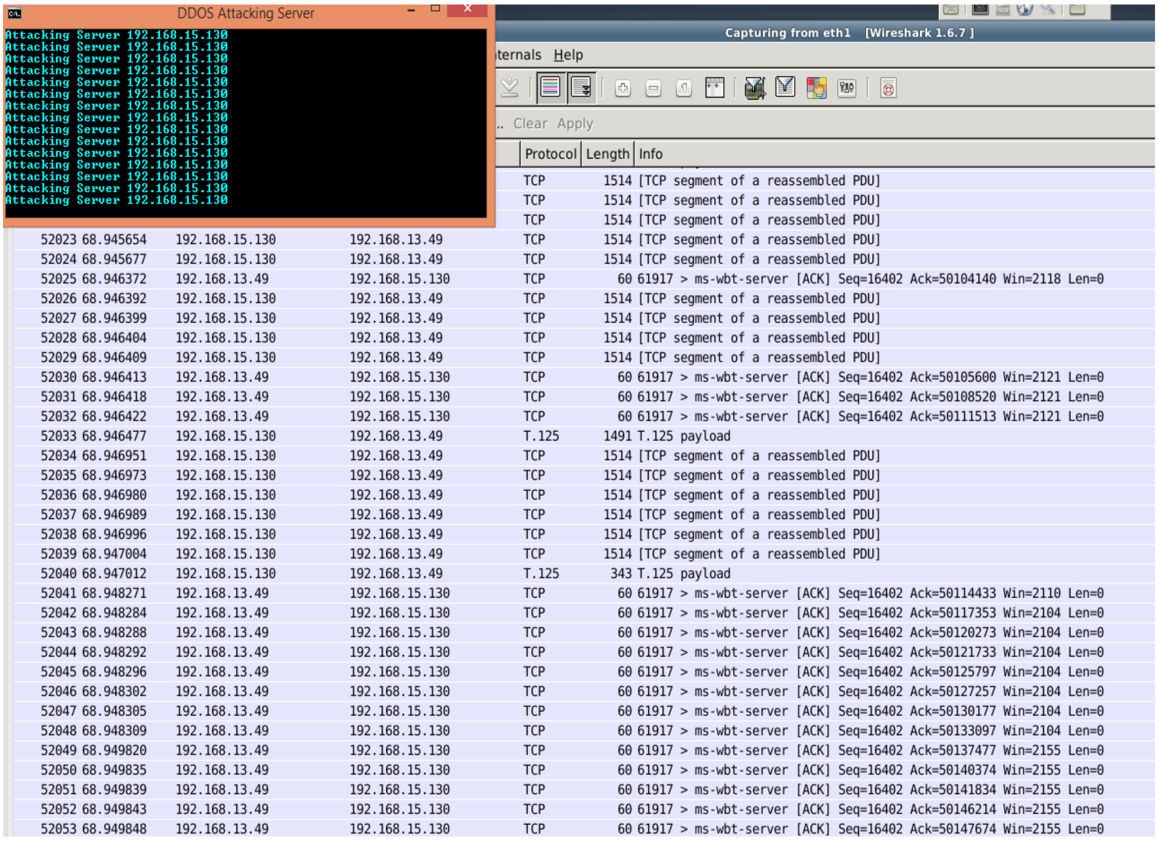

Figure 4.16: TCP SYN Detection Using Wireshark

flood.png

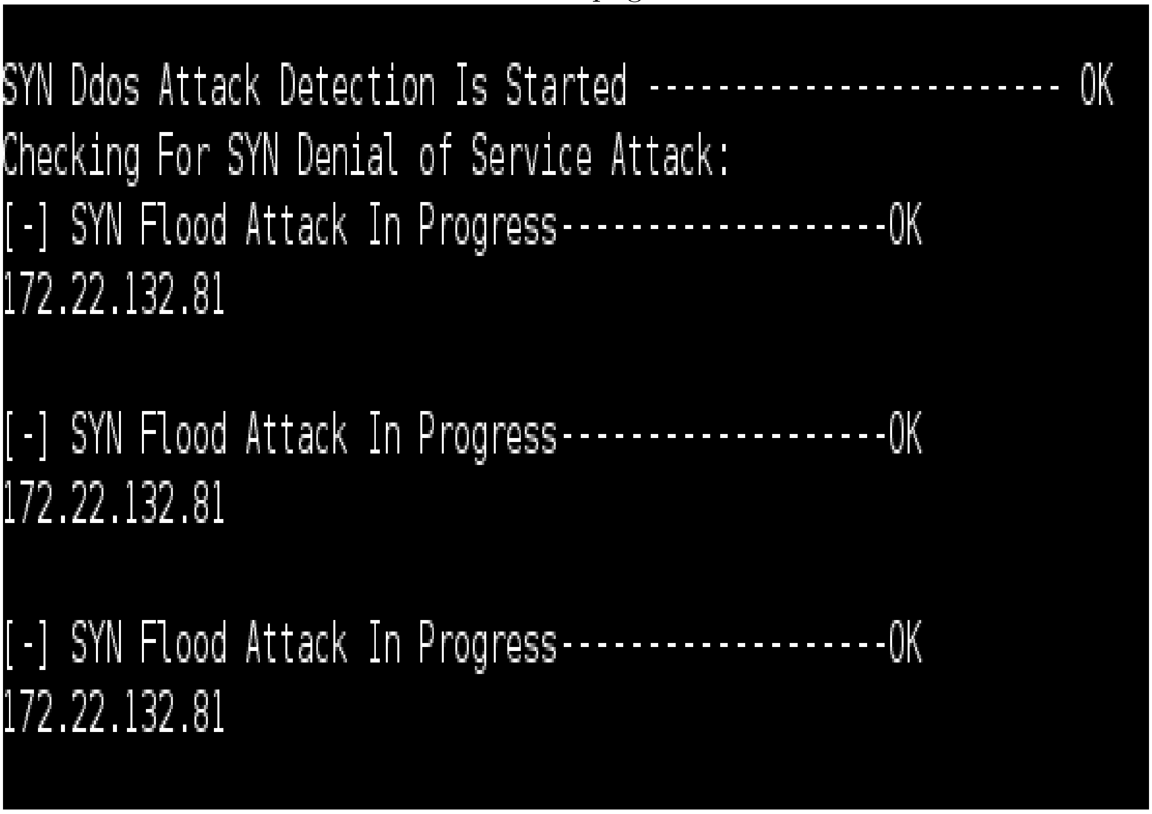

Figure 4.17: TCP SYN Detection Using Rule Based System

Figure 4.18: CPU Metrics Using Ganglia

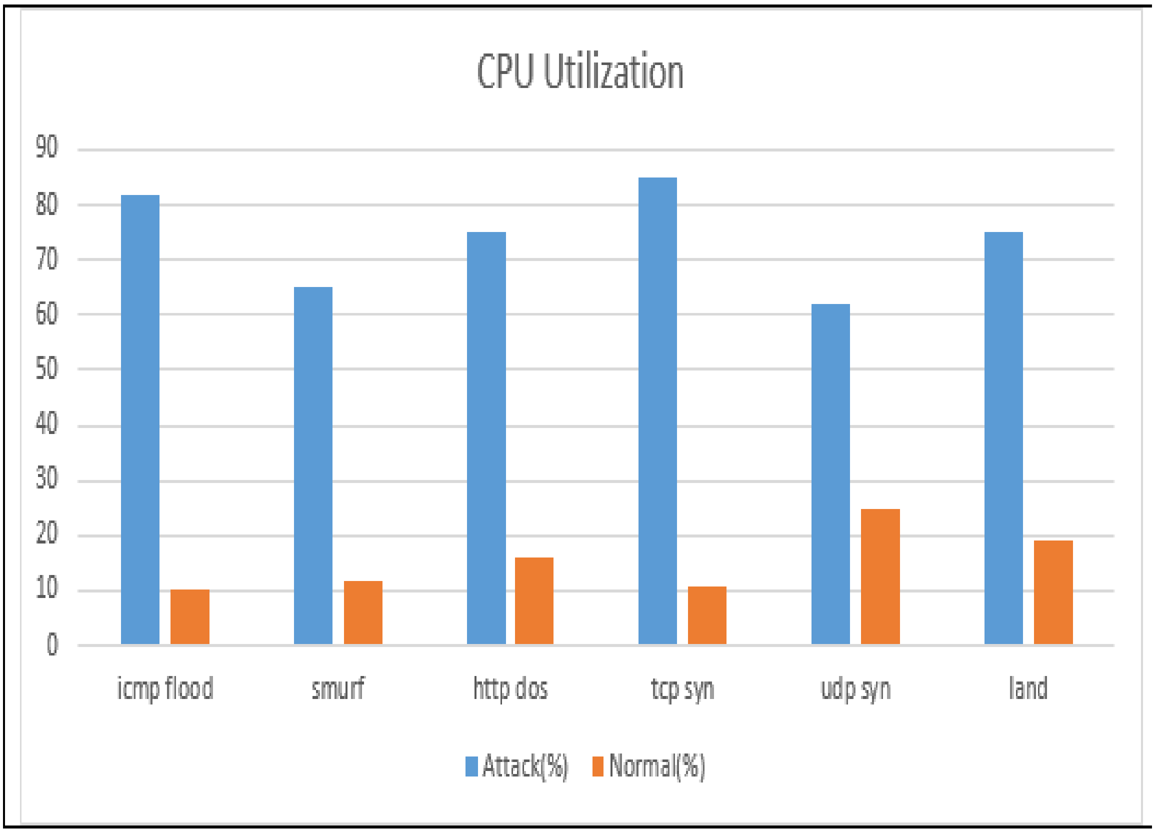

Figure 4.19: Comparison of CPU Utilization among attacks

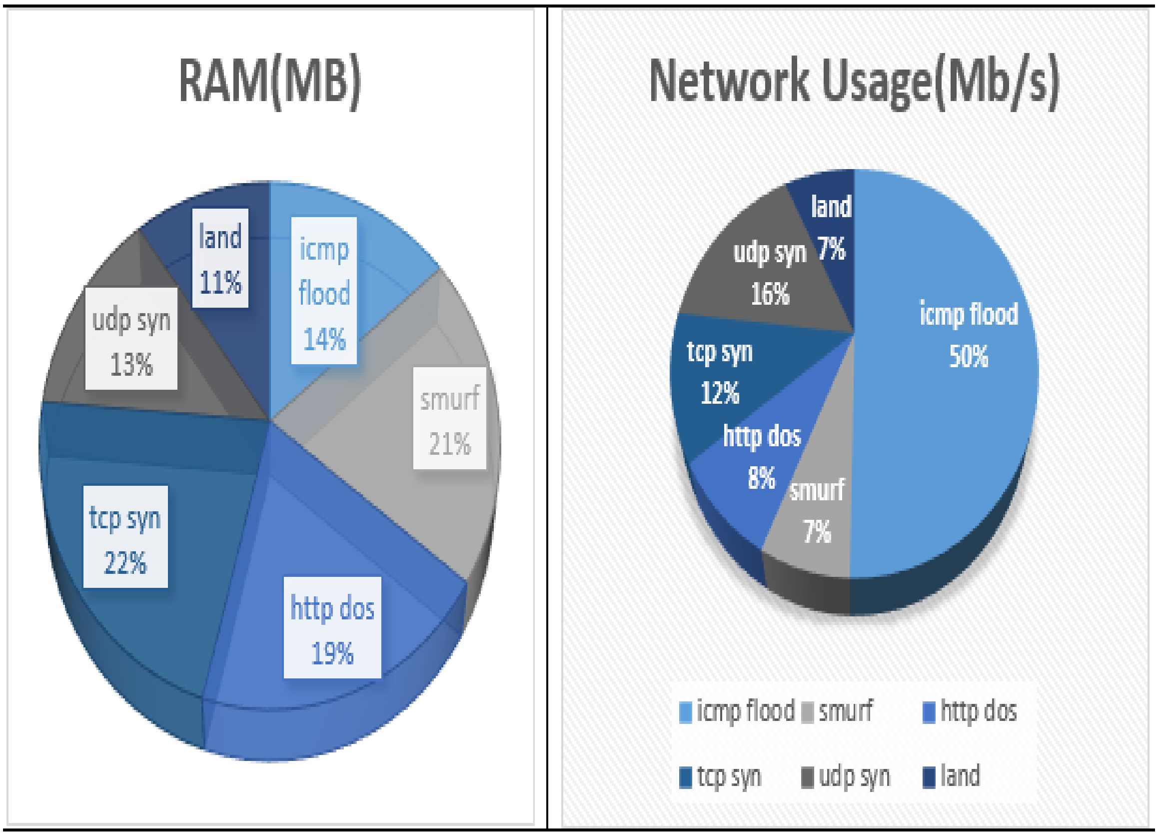

Figure 4.20: Comparison of RAM and Network usage among attacks

Chapter 5

Conclusion and Future Scope

5.1 Conclusion

Knowledge base approaches generally go for expert rules for detection of attack . This expert rules determine the severity of attack . Since this rules are dependent on human intervention efficiency is completely dependent on it. This rules database is updated and based on that its efficiency is increased. Since manual updation is required its efficiency is obviously less than that of artificial intelligence techniques.

We have proposed an Intrusion detection system for DoS attack in cloud so that there can be minimization of cyber-attacks. To conclude we have tried to reduce impact of dos attack by detecting it at initial state with improved accuracy so that accordingly actions can be taken. With this increasing number of cyber threats, it is necessary to detect such threats and accordingly actions shall be taken.

we have focused on identifying sources that facilitated with numerous characteristics but were exploited by Intruder. Since DDoS is major threat to cloud its detection is very challenging. An Hybrid Technique which is combination of Rule based Detection and Snort has been used for identifying the attacks. After Performance Comparison of Various types of DDoS attacks, it is concluded that TCP SYN Attack is more severe compared to other attacks . A hybrid technique is used for detection of attacks. The Rule based detection techniques works efficiently in cloud. With the several comparisons it is found that TCP SYN is more severe compared to various other DDoS attacks. This Rule based detection consists of IP detection,reducing redundant IPs and blacklisting them.

5.2 Future Work and scope

Automation of Monitoring and scanning the network by monitoring cpu usage , memory usage , log files. Most important of all its prevention by limiting the rate and blocking ip with maximum number of request.Even variations of attacks can be detected by updating Black list.

As I've mentioned earlier in proposed system that it is used to detect the dos attack and its severity.It can be extended by combing with different techniques for detection and alert generation. Even , Intrusion Response System and Intrusion Prevention System can be implemented making cloud more secure. After implementing the existing system one can enhance the security in cloud not just by being reliable on routers , switches , firewall and anti-virus. Graph based algorithm is yet to implement for large datasets of attacks and its severity along with its impact on network.

Bibliography

[1] Zhengbing H, Jun S,Shirochin VP Praveen Kumar Rajendrana*, B.Muthukumarb, G.Nagarajanc , An Intelligent Hybrid Intrusion Detection System for Private cloud: A systematic approach.

[2] Choo K-KR,"The cyber threat landscape:challenges and future search directions".Computers and Security 2011;30:71931.

[3] Chirag Modi, Dhiren Patel ,Bhavesh Borisaniya , Hiren Patel , Avi Patel MuttuKrishnan Rajarajan "A Survey of Intrusion Detection Techniques in cloud."

[4] Ahmed Patel a,b , Mona Taghavi a,* , Kaveh Bakhtiyari a , Joaquim Celestino Junior c . An Intrusion detection and Prevention system in cloud computing: A Systematic Review.

[5] Rup Kumar Deka a,* , Kausthav Pratim Kalitaa , D.K Bhattacharya a , Jugal k .Kalitaa .Network Defense : Approaches , Methods and techniques.

[6] Mazhar Ali, Samee U. Khan, Athanasios V. Vasilakos. Security in cloud computing : opportunities and challenges.

[7] Rashmi V. Deshmukha, Kailas K. Devadkarb Understanding DDoS Attack and Its Effect In Cloud Environment.

[8] JABEZ Ja, Dr.B.MUTHUKUMARb* . Intrusion Detection System (IDS): Anomaly Detection using Outlier Detection Approach.

[9] Seyed Mojtaba Hosseini Bamakana,b, Behnam Amiric, Mahboubeh Mirzabagherib, Yong Shia,b,d* . A New Intrusion Detection Approach using PSO based Multiple Criteria Linear Programming.

[10] Jianhua Chea*, Yamin Duanb, Tao Zhanga, Jie FanaStudy on the security models and strategies of cloud computing.

[11] Kaniz Fatemaa,*, Vincent C. Emeakarohaa, Philip D. Healya, John P. Morrisona, Theo Lynnb. A survey of Cloud monitoring tools: Taxonomy, capabilities and objectives.

[12] Hung-Jen Liaoa, Chun-HungRichard Lina,n, Ying-Chih Lina,b,Kuang-Yuan Tunga.Intrusion detection system: A comprehensive review.

[13] Roberto Di Pietro , Luigi V. Mancini , Intrusion Detection Systems , Springer ,Pg-192.

[14] Dimitrios Zissis*, Dimitrios Lekkas Addressing cloud computing security issues,Pg-5.

[15] S.V. Narwane, S. L. Vaikol Intrusion detection system in cloud computing environment.

[16] Akash G Mohod, Satish J Alaspurkar Analysis of IDS for cloud computing.

[17] S.Subashini*,V.Kavitha A survey on security issues in service delivery models of cloud computing.

[18] C. Mazzariello, R. Bifulco and R. Canonico, Integrating a Network IDS into an Open Source Cloud Computing Environment.

[19] Snehal G. Kene, Deepti P. Theng A Review on Intrusion Detection Techniques for Cloud Computing and Security Challenges.

[20] online Available at : https://usa.kaspersky.com/internet-security-center/definitions/smurf-attack.VmbacL9KUaA.

[21] Chia-Mei, Chen, Guan, D. J., Yu-Zhi, Huang, Ya-Hui, Ou. (2012, 9-10 Aug. 2012). Attack Sequence Detection in Cloud Using Hidden Markov Model. Paper presented at the Information Security (Asia JCIS), 2012 Seventh Asia Joint Conference on.

[22] Anteneh Girma, Moses Garuba ,Jiang Li ,Chunmei Liu Analysis of DDoS Attacks and an Introduction of a Hybrid Statistical Model to Detect DDoS Attacks on Cloud Computing Environment.

[23] R.Vanathi,S.Gunasekaran "Comparison of Network Intrusion Detection System in cloud computing environment."

[24] Online[available at] http://mayank-grover.me/denial-services-dos-attacks/

[25] Haris, S. H. C., Ahmad, R. B., Ghani, M. A. H. A. (2010, September). Detecting TCP SYN flood attack based on anomaly detection. In Network Applications Protocols and Services (NETAPPS), 2010 Second International Conference on (pp. 240-244). IEEE.

[26] Ohsita, Y., Ata, S., Murata, M. (2004, November). Detecting distributed Denial-of-Service attacks by analyzing TCP SYN packets statistically. In Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE (Vol. 4, pp. 2043-2049). IEEE.

[27] Wang, H., Zhang, D., Shin, K. G. (2002, June). Detecting SYN flooding attacks. In INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE (Vol. 3, pp. 1530-1539). IEEE.

[28] Durairaj, M., Persia, A. (2014). ThreV-An Efficacious Algorithm to Thwart MAC Spoof DoS Attack in Wireless Local Area Infrastructure Network. Indian Journal of Science and Technology, 7(5), 581.

[29] Korczyski, M., Janowski, L., Duda, A. (2011, June). An accurate sampling scheme for detecting SYN flooding attacks and portscans. In Communications (ICC), 2011 IEEE International Conference on (pp. 1-5). IEEE.

[30] Divakaran, D. M., Murthy, H. A., Gonsalves, T. A. (2006, September). Detection of SYN flooding attacks using linear prediction analysis. In Networks, 2006. ICON'06. 14th IEEE International Conference on (Vol. 1, pp. 1-6). IEEE.

[31] Haris, S. H. C., Ahmad, R. B., Ghani, M. A. H. A., Waleed, G. M. (2010, December). TCP SYN flood detection based on payload analysis. In Research and Development (SCOReD), 2010 IEEE Student Conference on (pp. 149-153). IEEE.

[32] Parasa Harika, Mrs D Raaga Vamsi (2012).Detecting and Alerting Tcp-Ip Packets against TCP SYN Attacks. International Journal of Computer and Organizational Trends,2(5).

[33] Rani, D. D., Krishna, T. S., Dayanandam, G., Rao, T. V. (2013). TCP Syn Flood Attack Detection And Prevention. International Journal of Computer Trends and Technology (IJCTT), 4(10), 3412.

[34] Rana, D. S., Garg, N., Chamoli, S. K. (2012). A Study and Detection of TCP SYN Flood Attacks with IP spoofing and its Mitigations. International Journal of Computer Technology and Applications, 3(4).

[35] Siris, V. A., Papagalou, F. (2006). Application of anomaly detection algorithms for detecting SYN flooding attacks. Computer communications, 29(9), 1433-1442.

[36] Thing, V. L., Sloman, M., Dulay, N. (2007, November). Enhanced TCP SYN attack detection. In IEEE/IST Workshop on Monitoring, Attack Detection and Mitigation (MonAM).