Attacks on Cloud and its Effects on IaaS

Submitted By

Namrata Patel

16MCEI15

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2018

Attacks on Cloud and its Effects on IaaS

Major Project

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology in Computer Science and Engineering(Information & Network Security)

Submitted By

Namrata Patel

(16MCEI15)

Guided By

Prof. Vivek Kumar Prasad

तमसो मा ज्योतिर्गमय

NIRMA
UNIVERSITY

INSTITUTE OF TECHNOLOGY

NAAC ACCREDITED ‘A’ GRADE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2018

Certificate

This is to certify that the major project entitled **"Attacks on Cloud and Its Effects on IaaS"** submitted by **Namrata Patel (Roll No: 16MCEI15)**, towards the partial fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Engineering of 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 major project part-I and part-II, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Prof Vivek Kumar Prasad
Guide and Assistant Professor,
CE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Sharda Valevati
Associate Professor and Coordinator,
CE Department
Institute of Technology,
Nirma University, Ahmedabad

Dr. Sanjay Garg
Professor and Head,
CE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Alka Mahajan
Director,
Institute of Technology,
Nirma University, Ahmedabad

Statement of Originality

I, **Namrata Patel**, Roll. No. **16MCEI15**, give undertaking that the Major Project entitled "**Attacks on Cloud and Its Effects on IaaS**" submitted by me, towards the partial fulfillment of the requirements for the degree of Master of Technology in **Computer Science & Engineering(Information & Network Security)** 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: May, 2018
Place: Ahmedabad

Endorsed by
Prof. Vivek Kumar Prasad
(Signature of Guide)

Acknowledgements

It gives me immense pleasure in expressing thanks and profound gratitude to **Prof. Vivek Kumar Prasad**, Assistant Professor, Computer Science Department, Institute of Technology, Nirma University, Ahmedabad 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.

It gives me an immense pleasure to thank **Dr. Sharda Valiveti**, Associate Professor, Computer Science Department, Institute of Technology, Nirma University, Ahmedabad for her valuable guidance and continual encouragement throughout this work.

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.

A special thank you is expressed wholeheartedly to **Dr. Alka Mahajan**, Hon'ble Director, Institute of Technology, Nirma University, Ahmedabad for the unmentionable motivation she has extended throughout course of this work.

I would also thank the Institution, all faculty members of Computer Science & Engineering Department, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

- Namrata Patel
16MCEI15

Abstract

In the era of Digitization and the e-commerce, all users are maintaining their data in a cloud and their personal information like documents, images, card information for quick payment and confidential data resides in the cloud. For Decades, most of the IT-industries, Government Agencies, Defense Sectors, Education hub, Hospitals and many more are using cloud services. One essential thing is that you require a network connection in order to get services from the cloud. There are many possibilities of attack in the cloud as the cloud is open and is connected to the internet. It means cloud run base on internet connection so chances of attacks on clouds are much more. Cloud is a hub of resources like network, storage, server, memory, CPU and any user uses these facilities worldwide at any time by subscription system. Cloud Computing provides so many services basically Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS) where an attacker always tries to breach the security at an IaaS level. Denial-of-Service (DoS) and Distributed- Denial-of-Service (DDoS) attacks on IaaS which target resources like CPU, Memory, RAM usage and this is the major concern about the Cloud Service Provider (CSP). In this research at an initial level, Intrusion Detection System (IDS) technique is been deployed to detect the DoS and DDoS attack, for that Snort has been deployed. To overcome the problem of DoS and DDoS attack, Reinforcement Learning (RL) algorithm has been used to train the Cloud behavior for the classification of an attack. The basis on Reinforcement Learning Policy it shows either attack is Normal or High according to that it will gives Positive and Negative Reward. If an attack has been Normal then CSP go for Prevent that attack otherwise go for Load Balancing.

Abbreviations

CC	Cloud Computing
CSP	Cloud Service Provider
IaaS	Infrastructure as a service
PaaS	Platform as a service
SaaS	Software as a Service
SLA	Service Level Agreement
DoS	Denial of Service
DDoS	Distributed Denial of Service
RL	Reinforcement Learning
MDP	Markov Decision Process
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
VM	Virtual Machine
LB	Load Balancing
EC2	Elastic Cloud Computing
EDoS	Economic Denial of Sustainability
CFU	Cloud Fusion Unit
DST	Dempster-Shafer Theory
FTA	Fault Tree Analysis
ACL	access control list
SVM	Support Vector Machine

–

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Cloud Computing

Cloud Computing(cc) is a hub of resources [1] (server, application, network, storage), where the user can use those resources at anywhere and anytime by pay per use system [2].One prerequisite is that you need an internet connection so as to get services from the cloud. This means that you need to take a look at a particular record or document you have housed in the cloud, you should first set up an internet connection [3].Their are many possibilities of attack in the cloud [4] as the cloud is open and is connected to the internet.It means cloud run base on internet connection so chances of attacks on clouds are much more. So this paper mainly focuses on different attacks on cloud computing environment and that attack which consume resources of cloud-like memory, CPU, RAM etc [5].Their are 3 types of cloud: 1) Public-cloud 2) Private-Cloud 3) Hybrid-Cloud.

A public-cloud is specified as being available from a 3rd party service provider via the web [6] and is a profitable way utilize IT solutions, remarkably for small-scale or medium-scale businesses. Google Apps [7] is an obvious example of a public cloud that is used by several organizations of all sizes of business.[cloud] A private cloud is used in a much secure way as compare to the public cloud [8], as the cloud is being elastic and services based on nature, and is handled by organizations. Private clouds provide greater control over the cloud infrastructure.

A community cloud [9] is regulated and used by a class of management that have mutual interests, such as particular security condition or a common mission. Finally, The cloud framework,which is a combination of two clouds (private, public) is called as hybrid cloud [10].Commonly, non-critical data is deployed to the public-cloud, while business-analytics benifits and data are kept within the authority of the institution.

1.2 Cloud Service Models

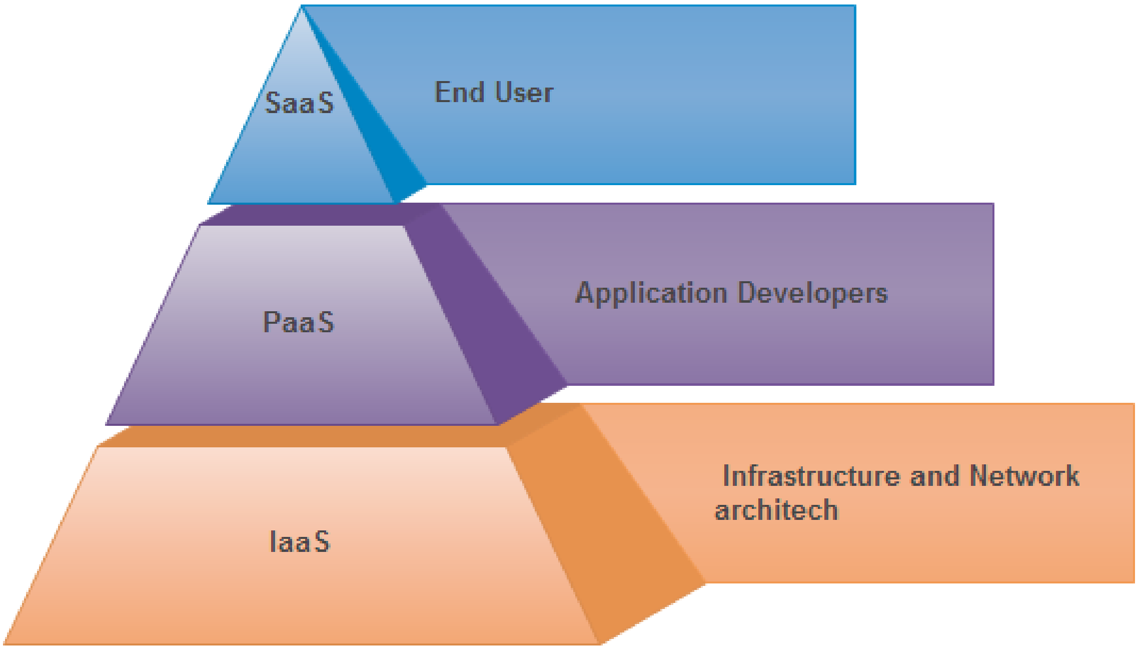


Figure 1.1: Cloud Service model

Cloud model provides three offerings:-

1.Software-as-a-Service (SaaS): The capacity gave to the users is to utilize the applications running on a cloud Infrastructure. The applications are available from different users gadgets through a thin client interface, like a web program (e.g., electronic email).

2.Platform-as-a-Service (PaaS): User deploys their Own developed Application, tools, or installing any platform or running any platform on cloud infrastructure [11]. PaaS delivers platform layer resources, as well as operating system support and application, build out a framework that can be using into building higher-level services.

3.Infrastructure-as-a-Service (IaaS): In this model, cloud service provider provides users to deliver processing capacity, network bandwidth, storage capacity and other underlying resources.

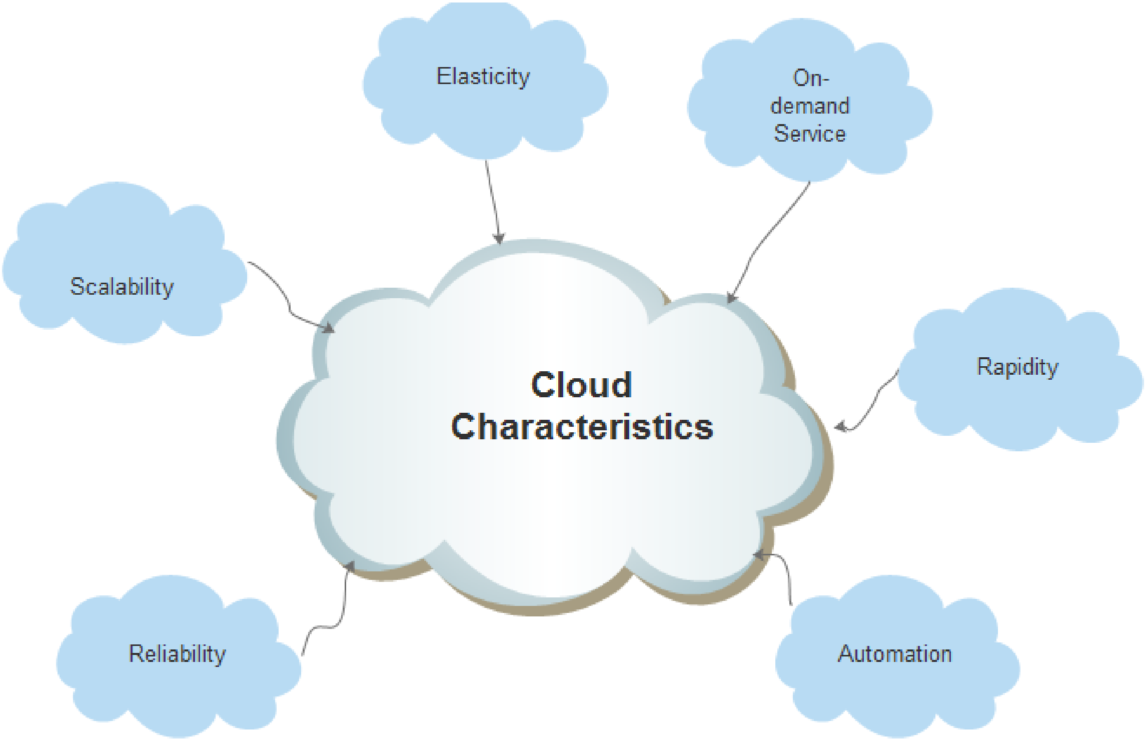


Figure 1.2: Attacks on service model

Cloud has many characteristics [12] like 1)flexibility/elasticity 2) scalability of infrastructure 3) broad network accessed 4) location independence 5) reliability 6) cost-effectiveness 7) sustainability.

1.3 Major Security Concern in Cloud Computing

In cloud mainly three level are their, have different security concern at different level. For example IaaS Level mainly contain following types of attack connection like flooding,DDoS,Network attacks, Hardware Interruption,Hardware Theft, Hardware Modification, Misuse of Infrastructure, Natural Disaster [13].At PaaS Level the major security concern are software modification, software interruption,alteration of data, session hijacking,traffic flow analysis. At SaaS level key issues are interception, data interruption, privacy breach, session hijacking, traffic flow analysis and impersonation.

Major Security Concern		
No.	Types Of Attack	Description
1.	DoS(Denial of Service)	Both attacks are DoS and DDoS are denial-of-service and distributed denial of service respectively. The assaults work by asking for such a significant number of assets from a server that the server can't react to authentic solicitations. A DoS is an assault that starts from a solitary gadget. A circulated DoS (or DDoS) includes malevolent movement from different gadgets.[14]
2	Interception	A block attempt implies that some unapproved party has accessed a benefit . The outside intruder can be any program, any human or any computer system.
3	Data Interruption	In Data Interruption, benefits of system may be deleted, or become lost, or unreachable or unworkable.
4	Privacy Breach	At SaaS level presents various lawful difficulties towards security issues included in information store in numerous areas in the cloud, and also increasing the risk of privacy breaches.
5	Session Hijacking	At the point when a client sign on to a web services, for example, the cloud, a session was established. This client session monitors client data, including a session ID, to confirm client demands for information.
6	Traffic Flow Analysis	It analyze traffic to create a congestion for leading down the cloud.

Table 1.1: Security Concern

In this research,we are going to see which attacks are going to affects on the IaaS(Infrastructure as a service) level of cloud computing and one of those attack is DDoS(Distributed Denial-of-Service) attack, where the attacker consumes Data Center servers memory, CPU and RAM .To detect these attack by using IDS Intrusion detection system (IDS).the Reinforcement Learning techniques has been used to learn the cloud behaviour for the classification of attack.

1.4 DoS and DDoS Attack in Cloud

A (DoS)denial of service is kind of attack were the security matters and this takes place when an attacker takes an action that averts legitimate client from acquiring targeted cloud systems, devices or other cloud resources. DoS and DDoS are still the major challenges in the cloud computing environment[15] . In Cloud Computing, these assault crushes cloud servers by purposely infusing malicious packets on to the cloud to quickly eat up cloud resources like VM(Virtual machine), servers and the resources such as CPU, memory, and RAM [16].

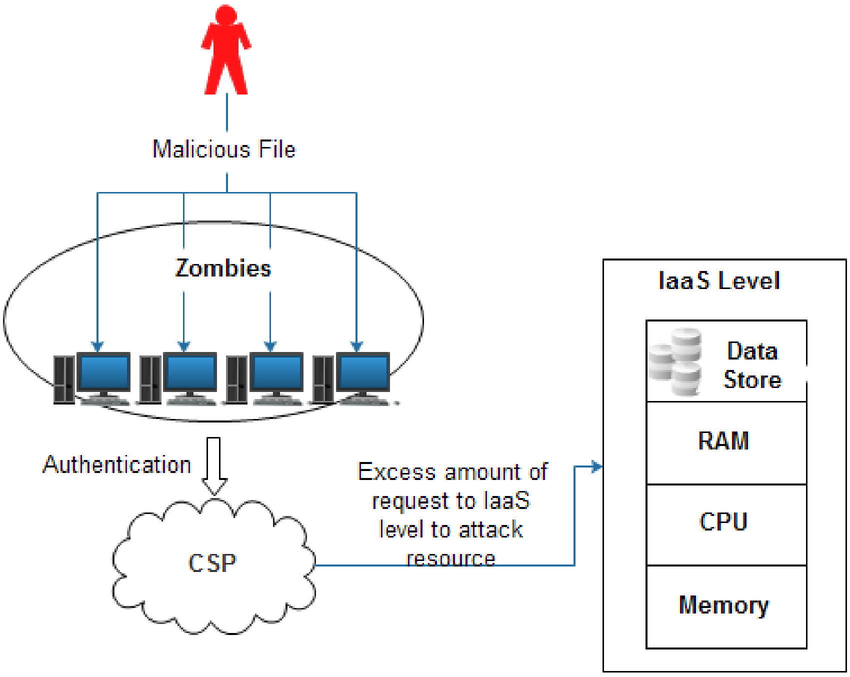


Figure 1.3: Flow of attack at IaaS level

In the above mentioned diagram attacker will hide its malcious file in the form of adware or anyother way to insert its malicious code into the victim computer for this attacker will use javascript to create malcious file or anyother scripting language.By running this malcious code victim computer indirectly become zombiee in the network.Attacker will pass all its request through the zombiee network by which it will help him to hide in the network.This zombiee network will send multiple user request to the clodlet which will cause the network to get into the halt state and sometime system crash.This will let the system to perform load balancing and sometime for better service it have to implemented fedration technique which will increase the cost of the cloud for service provider.

1.5 Various Example of Attack Pattern

SYN Flood Attack: When client(a system) established any connection like TCP connection to the server which provide a service, at that time client server transfer an order of messages for establishing the connection. These type of connection establish method utilizes for all kind of TCP connection likewise telnet connection, Web connection or emailing connection, etc. SYN message sending by a client to server, asking to server to establish a connection. After that server gives acknowledgement for that SYN message by sending a SYN ACK message to the client side. By this kind of establishment the client and server is then open, and they can exchanged data between them. In this once server acknowledge client request at that time client should acknowledge for obtaining a connection. After this some times later, the system may crash, it may fatigue memory, or be concluded else way unworkable.

TCP Flooding: Apparently, In casse of TCP flooding attack , we do discussed about the TCP SYN flood attack also. In TCP ACK Flood, many of packets who's types are in TCP,are send the to victim to exploit its system and network assets. All this build upon the Operating system , an open and close port might reply a TCP packets, which aim more congestion and workload on to the cloud and cloud resources. There are more other, such as NULL flood and Reset TCP(RST) flood.

UDP Flooding: In UDP Flooding attack, attacker send many UDP packets to arbitrary port on to the cloud systems. At that time victim system get an UDP packets, and this will assume that an application is waiting on the destination port. When system recognize that their is no or any application which remains on the port, at the same time this will also generate ICMP packets of destination to the fake source address and at the same time sufficient UDP packets are sent to the ports on victim's systems, and as a result the system will go down.

ICMP Flooding: This is like other flooding attacks, which are sends on bunch of packets to victim's system as in ICMP flooding, this broadcast group of ICMP packets, which are generally a ping request.

Chapter 2

Literature Review

- In this paper [17] authors uses DDoS detection technique in to the cloud where their solution is deploying IDS(intrusion detection system) technique to every single VM(virtual machine) with data fusion method in frontend and when DoS attack happens, the IDS will generate the alert,which is stored into the database and placed within Cloud Fusion Unit(CFU). For that they use Dempster-Shafer Theory(DST) and Fault Tree Analysis(FTA).

- In this paper [18] authors find threshold calculation mechanism which detect the DDoS attack perform or not.But in this proposed system, they only shows the static threshold calculations and numerals at an application level.

- In this paper [19] author focused on profile based network intrusion detection for the purpose of keeping security in the cloud system. The perception of making a virtual machine profile DB(database) that will define the attack pattern and the profile will be updated according to the new attack patterns and mesh attack patterns frameworks by anomaly detection system. Based on different incoming and outgoing packets traffic are observed, the anomalous behaviour of system or traffic flow on the specific profile of appropriate virtual machine will be noted down.

- In this paper [20] author shows comparative analysis of different DDoS detection, mitigation, and defending methods. They shows different tools for DDoS attack and performance metrics. This detection and secure standards using learning algorithms for protecting cloud support system.Anomaly detection technique can be used to defend zero day attack.

- In this paper [14] author shows different types of DoS attacks description and recent technique to prevent attacks using black holing and router ACL(access control list), firewall,intrusion detection system, and intrusion prevention system, using signature based and anomaly based detection. They defines only machine learning techniques like ANN(artificial Neural Network),SVM (support vector machine), genetic algorithm, fuzzy logic etc.

- In this paper [21] they exhibit the cloud security challenges at the correspondence level (between clients what's more, cloud, correspondence happening inside cloud foundation), for Virtual machines. They examine different methodologies proposed in the writing to counter the security issues. Utilizing tables, they demonstrate the security highlights for every countermeasure plot.

- In this paper [22] they presents an investigation of the kinds of DoS attacks with the new attacks against virtual machines and hyper-visors in distributed computing condition. Moreover, the creators additionally list the undertaking system safeguard and distributed computing barrier against DoS.

- In this paper [23] their survey cloud is targeted by DDoS attacks. After that they categorize DDoS attack into an application level means in SaaS layer of cloud and an Infrastructure level means IaaS and admit different tools to organize these kind of attacks.

Paper	**Problem Identified**
[13]	They only detect DDoS attack and only deploy on Virtual Machines.
[18]	They only shows the static threshold calculations and numerals at an application level.
[19]	Focused on profile based network intrusion detection.
[20]	They shows different tools for DDoS attack and performance metrics.
[14]	Only defined Router Access Control List.
[23]	They categorize DDoS attack into an application level means in SaaS layer of cloud.

Table 2.1: Gap Identified by Proposed Model

Chapter 3

Problem Statment

3.1 DoS and DDoS on IaaS

The ratio of detecting and analyzing DoS attack is increasing ,by which DDoS become the most serious threat. There are 20% increase trade proclaimed DDoS attack's matter on their infrastructure. Throughout several threat, which include several less popular attacks which impact a huge consideration in the study area. Another remarkable threat guide is, Amazon EC2(Elastic Cloud Computing) resources encountered enormous DDoS attack. These attacks result in overwhelming downtime, business fall with various other losses and problems. A Verisign intense Security Intelligence Service record maximum threat target of ddos attacks is cloud and especially on SaaS(Software as a Service)and IaaS(Infrastructure as a Service). The main issue of DDoS attack in the cloud is "commercial loss". In this the creators address the direct business misfortune due to a DDoS assault to around 444k USD. Foundation as a Service (IaaS) mists run customer benefits inside Virtual Machines (VMs).

In cloud virtualize server is the major component for the elastic cloud and on-demand capacity of the cloud is also an important issue, where VMs takes extra resources when required and restored unused resources when the cloud is in idle condition. Cloud Computing's overwhelming selection drift is expected to the on-request computing and asset accessibility capacities. This limit enables the cloud structure to give huge resources by scaling, as and when their is a need on a VM. Along these lines, a VM won't experience an advantage power outage as a sufficient measure of on-ask for resources are available in the cloud. This component of " versatility " or "auto-scaling" comes to fruition into money related misfortunes based DDoS assault which is known as Economic Denial of Sustainability (EDoS) assault or Fraudulent Resource Consumption (FRC) assault[24].

In our research domain, we see which attacks have their effects on IaaS(Infrastructure as a service) resources and those attacks are DoS(Denial-of-Service) and DDoS(Distributed Denial-of-Service) attacks which consume Data Center's Memory, CPU and, RAM. Detection of this attack by using IDS(Intrusion detection system) using Snort. If an attack is not in under control then Load Balancing technique has to be used.

In DDoS assault the aggressor abuse the "pay-as-you-go" show. Aggressors by and large plant bots and trojans on the pacify of machines over the internet and target web organizations with Distributed Denial of Service assault.

DDos takes the state of an Economical DoS attack when the victim benefit is facilitated in the cloud. Associations exist (otherwise called "Booters"), which give a system of bots to their buyers to design the DDoS attack on their adversary sites. Thought processes of these attacks extend from business rivalry, political competition, fraud to cyber wars among nations. The cloud worldview gives tremendous open doors and advantages to purchasers and a similar arrangement of highlights are accessible and might be valuable for DDoS attackers. Attackers who design a DDoS attack would send enough phoney request to accomplish "Denial of Service". Be that as it may, this attacks would produce substantial attack usage on the victim server. "Auto-scaling" would take this "overloading" circumstance as input and include more CPUs (or different resources) to the dynamic pool of resources of this VM.

Once a VM gets conveyed, it begins as a "Normal load VM". Presently, let us expect that the DDoS attack has begun and the VM gets over-burden ("Overloaded VM"). The overloaded condition triggers auto-scaling highlights of cloud resource distribution, and it will pick one of the numerous procedures accessible in the writing for VM resource assignment, Migration of VM, and VM arrangement. Overloaded VM might be given some more assets or moved to a higher resource limit server or might be promoted by another server. On the off chance that there is no relief in set up, this procedure will continue by adding the resources. This circumstance may last till service provider can pay or CSP(cloud service provider) expends every one of the assets.

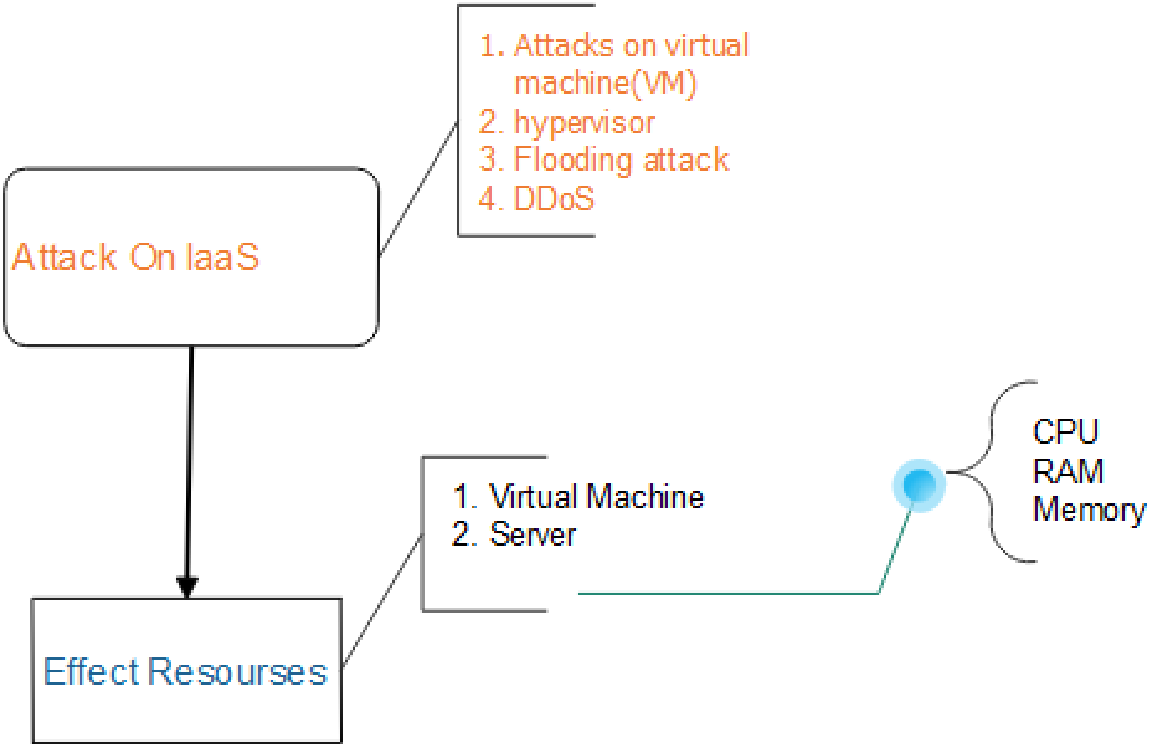


Figure 3.1: Attack on IaaS

At long as long, it will prompt to "Service Denial". This prompts on-request resource charging, and in this manner, economic losses far beyond the arranged spending plan may happen. One minor solution is to run VMs on settled or static asset profile where the SLA does not have any arrangement for extra assets on request. For this situation, the DDoS will specifically bring about "Denial of Service" and all the pleasant features of the cloud will be lost.

3.2 Attack on IaaS

As shown in figure 3.1 IaaS contains Resources, network, storage, virtual machine where there are attacks on VM, attacks on hypervisor, flooding attack, or DoS and DDoS attack, etc. when these attacks happens IaaS resources which are virtual machine and servers. So here we focused on DoS attack which consume many resources. DDoS attack is a big concern to the availability of resources. The big objective of a DDoS attack is to make the client or customer unreachable to cloud resources. Target could be web server, CPU, storage, and any other network resources. DDoS attack may diminish the completion of

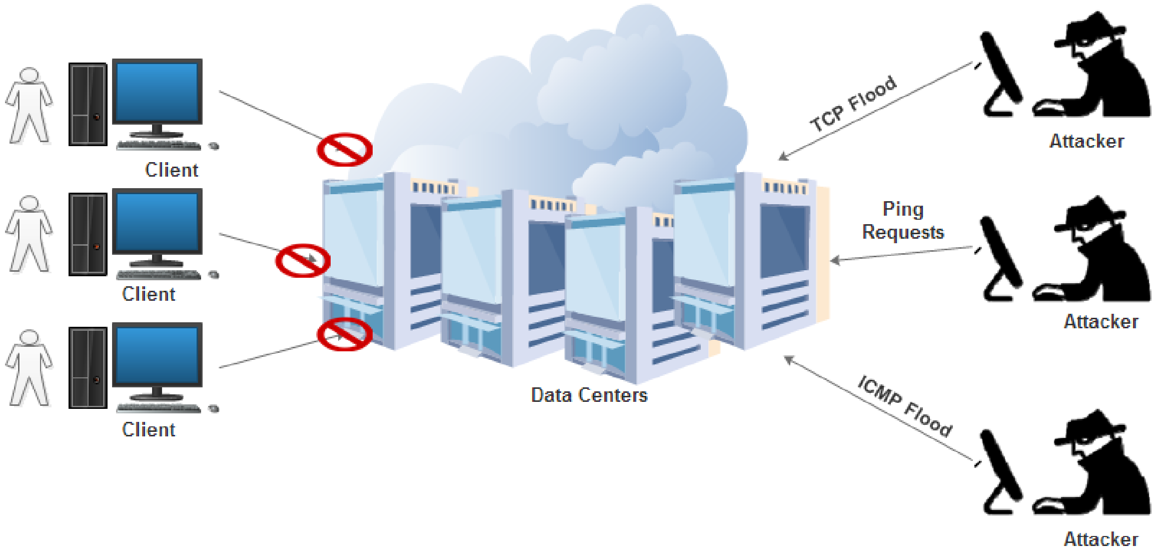


Figure 3.2: DDoS attack at Data Center

cloud services. Admit to cloud security alliance, DDoS is one of the top nine hazard to cloud computing.

As in diagram 3.2 Attackers make TCP Flood, ICMP Flood, UDP Flood to doing DoS and DDoS attack at the same time legitimate Client unable to access Data Center resources. Basically CSP not classify the request and users are either legitimate or not.

Table:3.1 gives information about CC trust system with different cloud parameters mainly IaaS. The table reflects various models. There is no single model that fulfils all the requirements of cloud architecture. Identity management, security in terms of the application and data both, SLA management[30] and heterogeneity[31] are the main models of cloud architecture which are not being fulfilled. Domain-based model is deployed on all three platforms of Iaas, PaaS, SaaS[32]. This model is tested using simulation and it supports the heterogeneity while Identity management, Data security[33], SLA support are not met. Model built on trusted platform service is deployed on IaaS and only proposed model is given which meets requirements like identity management[34], data security, and heterogeneity. While SLA support is not met. Both above models are not being implemented. Built on Trusted Computing Platform's prototype is implemented. While its support on IaaS platform only; where Identity management, Data security, SLA support and heterogeneity all are not met.

Work	Type	Identity mgmt & Authentication	Data Security	Cloud Layer	Implemented	Comments
[25]	Based on Domain	No	No	IaaS, PaaS, SaaS	No	Only Simulation testing has been done
[26][27]	Built on trusted platform service	Yes	Yes	IaaS	No	Model has been proposed , no real time implementation
[28]	Trusted Computing platforms has been used	No	No	IaaS	Prototype Implemented	Only prototype model has been used to proof
[29]	NA	Yes	No	IaaS, PaaS, SaaS	No	Only model has been identified

Table 3.1: Research Gap identified by Parameter

Chapter 4

Proposed Solution

Unavailability Resource in Cloud can be number of factors such as failure of cloud infrastructural factors or any software applications or tools. Distributed Denial of Service focused towards a cloud computing.

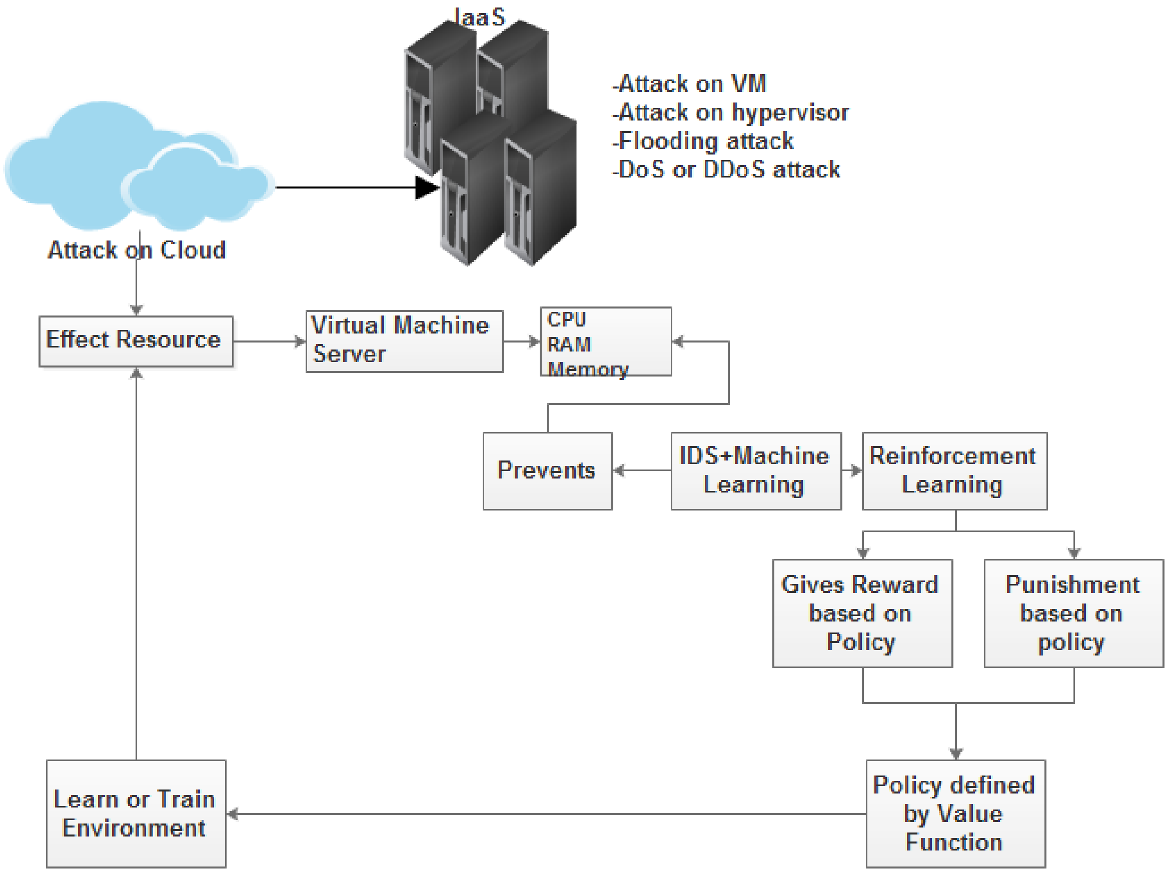


Figure 4.1: System Architecture Workfklow

As in a cloud various attacks are possible, one of the attack is at an Iaas Level. Which will effect the VM, Hypervisor, Flooding Attack, DoS and DDoS attack.For prevention of the attack we have implemented IDS as well as Machine Learning technique. In machine learning we have used Reinforcement Learning technique which will reward based on the policy and the punishments as well. Value function will help to learn and train the system.By the help of this we can decrease the attack at an IaaS level.

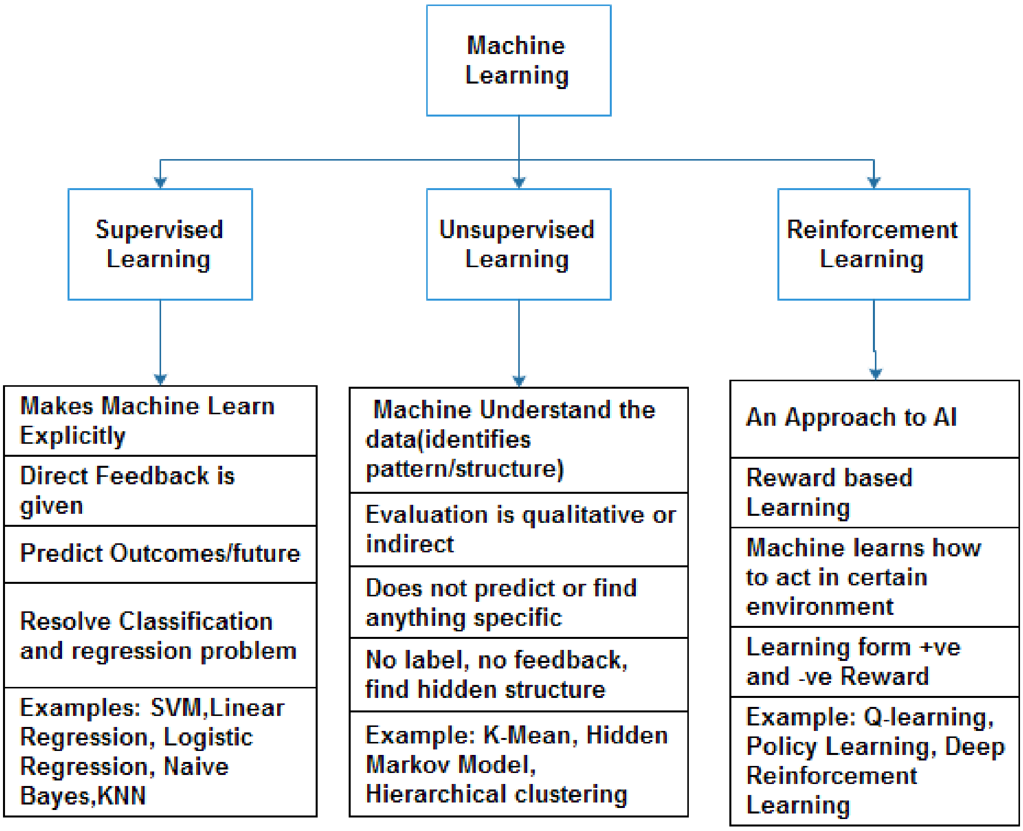

Machine Learning
Supervised Learning
Unsupervised Learning
Reinforcement Learning
Makes Machine Learn Explicitly
Direct Feedback is given
Predict Outcomes/future
Resolve Classification and regression problem
Examples: SVM,Linear Regression, Logistic Regression, Naive Bayes,KNN
Machine Understand the data(identifies pattern/structure)
Evaluation is qualitative or indirect
Does not predict or find anything specific
No label, no feedback, find hidden structure
Example: K-Mean, Hidden Markov Model, Hierarchical clustering
An Approach to AI
Reward based Learning
Machine learns how to act in certain environment
Learning form +ve and -ve Reward
Example: Q-learning, Policy Learning, Deep Reinforcement Learning

Figure 4.2: Types of Machine Learning

- Supervised Learning: Supervised learning is a way to deal with machine discovering that depends on preparing information that incorporates expected answers. A Artificial Intelligence(AI) utilizes the information to fabricate general models that guide the information to the right answer[35].

- Unsupervised Learning: In unsupervised taking in, an AI framework is given unlabeled, uncategorised information and the framework's calculations follow up on the information without earlier preparing. The yield is reliant upon the coded calculations[36].

- Reinforcement Learning: Reinforcement learning, in the frame of Artificial Intelligence(AI), is a sort of dynamic programming which is trains algorithms using a system of reward and punishment[37].

4.1 Reinforcement Learning in Cloud Computing

we are discussing about Reinforcement algorithm to detect and reduce the IaaS level attack in cloud computing.Mainly reinforcement learning(RL) agent receive data centre information from the cloud environment [38]. Modification in the environment or framework are denoted in various IaaS Resources to the agent are mapped into the different cloud states. Agent will perform the action and they observed the acknowledgement in the form of positive and negative reward.The Activity chosen depends on maximization of particular principal case of these standard are the quick reward,normal reward,per time step and total discontinued reward among other. In the last steps of the RL procedure, Agent map changes in the cloud environment as a new state and they update the learning policy to upgrade future rewards.

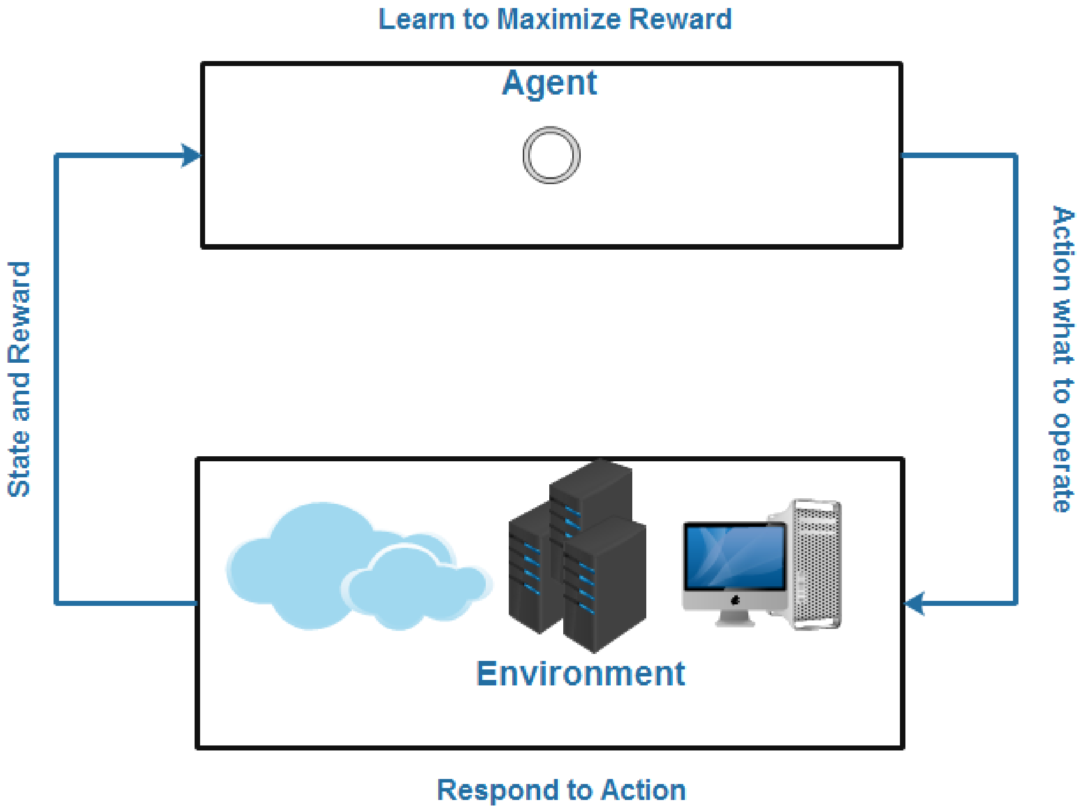


Figure 4.3: Intrusion Detection System

Markov Decisions Process Reinforcement Learning issue can be formally characterize as a Markov Decision Process(MDP).MDP is characterize in four tuple (S,A,R,T)

'S' is a limited set of arrangement by an agent.The state are mapping data from the agent sensors input. A will be limited arrangement of actions available to the agent to perform. R(c,a) characterize the reward in state 'c' after performing activity 'a'. T(c,a,c') $\longrightarrow [0,1]$ is a probability transition function. It characterize the probability to travel from state c to state c' after finished activity 'a'. In a MDP the R(c,a) and the T(c,a,c') functions just relay upon the present state and activities.

p$_r = \{c_{t+1} = c' | c_t, a_t, c_{t-1}, a_{t-1}, ..., r_1, c_0, a_0\}$
Eq(1) is an special case of the more broad situation where the present state additionally depend on past state and activity state.

$p_r = \{c_{t+1} = c' | r_{t+1} = | c_t, a_t\}$

when eq(1) is equal to eq(2) for all c', r and the sequence $c_t, a_t, r_t, a_1, r_1, c_0, a_0, r_0$ we say that the state the state have a markov property. The gives the capacity of the utilization of markov prpoerty in RL is extremly valueable it gives us predict the next state and normal reward with just the present state activity.

Value function

At each time step agents watch state and execute activity.The state-activity mapping is know as the policy π casually the probability of choosing an activity 'a' in a given state 's' under policy π is $\pi(c, a) \longrightarrow [0,1]$ and is a characterize by $V^\pi(c)$ is defined as eq(3)

$V^\pi(c) = E_\pi\{R_t | c_t = c\} = E_\pi\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | c_t = c\}$

Q Learning

The fundamental substance of Q-learning is that you have a portrayal of the natural states c, and conceivable activities in those states a, and you take in the estimation of each of those activities in each of those states. Instinctively, this esteem, Q, is alluded to as the state-activity esteem.

In Q-learning, you begin by setting all your state-activity esteems to 0 (this isn't generally the case, yet in this straightforward usage it will be), and you go around and investigate the state-activity space. After you attempt an activity in a state, you assess the express that it needs to prompt. In the event that it needs to prompt an unfortunate result you lessen the Q esteem (or weight) of that activity from that state with the goal that different activities will have a more noteworthy esteem and be picked rather whenever you're in that state. Also, in case you're remunerated for making a specific move, the heaviness of that activity for that state is expanded, so will probably pick it again whenever you're in that state. Vitally, when you refresh Q, you're refreshing it for the past state-activity mix. You can just refresh Q after you've seen what comes about.

Naturally, the adjustment in the Q-learning for playing out the activity an in state c is the contrast between the real reward (reward(c, a) + max(Q(c'))) and the normal reward (Q(c,a)) increased by a learning rate, alpha. You can think about this as a sort of Predictive control, driving your framework to the objective, which is, for this situation, the right Q-learning.

Here, we assess the reward of advancing when the small change is in two stages ahead as the reward for moving into that state (0), or more the reward of the best activity from that state.

There are different way to calculate the optimal policy an to amplify that what reward over the time can be acquired. One of the most generally utilize mechanism is Q learning. In Q-learning the agent repeatedly tries to estimate the value function.

$$Q(c,a) \longleftarrow (1-\alpha)Q(c,a) + \alpha\Big(r + \gamma max_a Q(c',a')\Big)$$

4.2 Intrusion Detection System in Cloud

IDS is a very convenient tool for reporting forensic data(evidence) that may be used in judicial proceedings if the trigger persone of a criminal breach is prosecuted.we can apply IDS at any level at cloud like at Virtual Machine(VM) itself or In hypervisor or host system or in virtual network.[39]

NIDPS(Network Intrusion Prevention system): it analyze network traffic of distinct network. Once detection of attack is done or any irregular behavior is detect, at that time alarm generate can be sent to administrator. And prevention system prevent that attack by some logics.[17]

HIDPS(Host Intrusion Prevention System): host based IDS technique deploy on as name suggest on the particular host to audit and inspect internal system behaviour. There are many types if IDS which are: Statistical approach the system examine the activeness of object(such as CPU usage or number of TCP connection) in term of analytical distribution and generate profile of such component which shows their behaviour.so, they making two profiles which are one is keep during training phase and another during the detection which is current profile of object.and if any difference between these two profiles are detect that means anomaly recognized. Detection Method: there are mainly two detection methods which are 1. Signature based intrusion detection and 2. Anomaly based intrusion detection in first one method attempts to define a set of rules that will use to define a given pattern match to the set of rule that it will attack. And in anomaly base detection identify abnormal behavior compare with normal behavior.

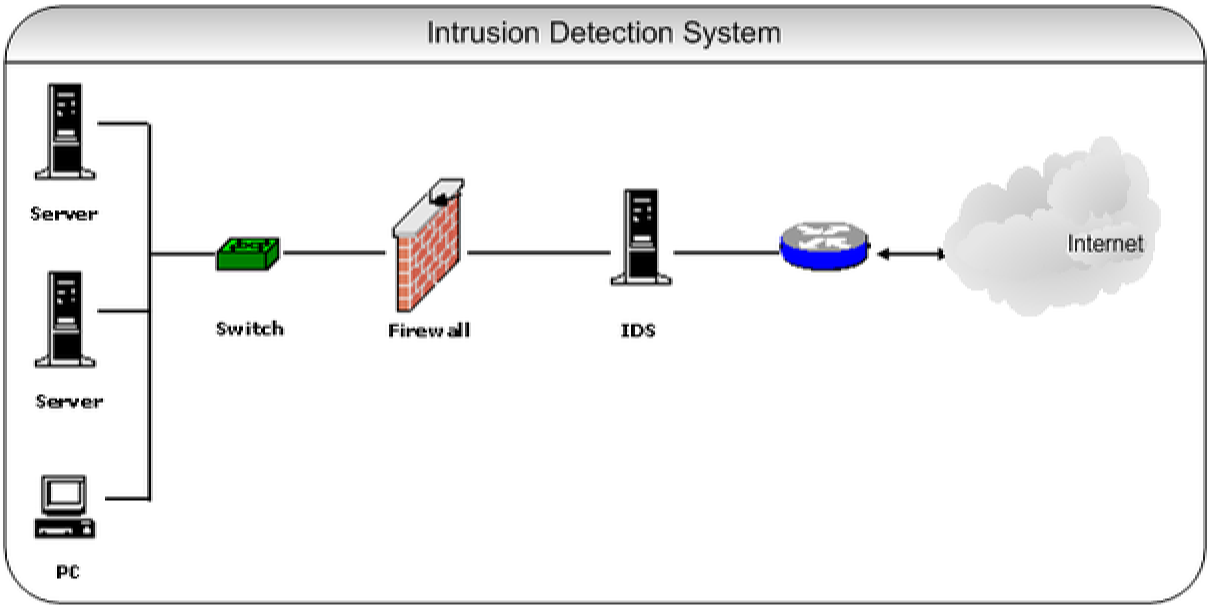


Figure 4.4: Intrusion Detection System

As shown in figure IDS placed between internet and the system which detect intruder by any IDS techniques. Here “Snort” is used for IDPS so we placed snort in between internet or cloud and system.

4.3 Proposed Algorithm

Algorithm 1: Proposed Approach

```
Initialize the user state
        s(c,a)$\longrightarrow s'(c',a')$
    If Updated User State gets reward / punishment
      if( s'(c',a') = (c$_t$, a$_t$)) *then*
          r = r + 1(reward)
      else
          r = r - 1(punishment)
After getting the reward the cases will be generated
      Value function of user-state
        switch(c',a')
      case 1:
          $\pi(c',a') = (0,0)$
          user a'= Normal
          break;
      case 2:
          $\pi(c',a') = (1,0)$
          user a'= Abnormal
          break;
       case 3:
          $\pi(c',a') = (0,1)$
          user a'= Moderate
          break;
       case 4:
          $\pi(c',a') = (1,1)$
          user a'= High
          break;
         Whether the rewarded user state is Abnormal / Normal
           If (user a' = (1,1) ) then
                s'(c',a') = Migrate
           else
                s'(c',a') = Handled User Request
```

As in algorithm initially User State "s" define state "c" and activity "a" where s(c,a) is either (0,0),(0,1),(1,0),(1,1). which is defined by policy in value function as defined early. After that basis on User State it will gives the reward or a punishment. After getting the reward the cases will be generated that saws the load is Normal, Abnormal, Moderate, or High. where User State (0,0) is Abnormal state and for that situation no solution to prevent DDoS at this level so we go for Migration technique.

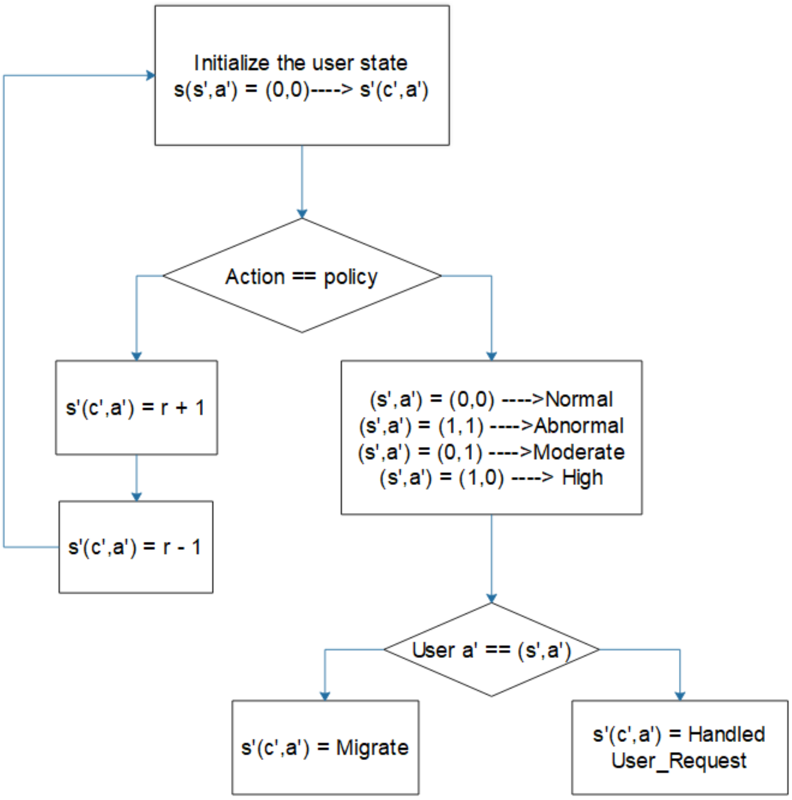

Initialize the user state
s(s',a') = (0,0)----> s'(c',a')
Action == policy
s'(c',a') = r + 1
s'(c',a') = r - 1
(s',a') = (0,0) ---->Normal
(s',a') = (1,1) ---->Abnormal
(s',a') = (0,1) ---->Moderate
(s',a') = (1,0) ----> High
User a' == (s',a')
s'(c',a') = Migrate
s'(c',a') = Handled User_Request

Figure 4.5: Flow Of Algorithm

Chapter 5

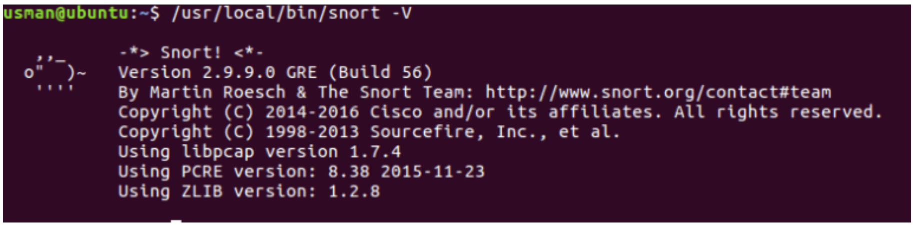
Implementation and Results

5.1 Tools and Technology and System Configuration

- Programming Language: Python (Version 3.6)
- Library / Platform: Anaconda, numpy, metploit
- IDE: Spyder
- Processor : Intel core i5– 2450M CPU @ 2.50 GHz
- RAM: 4 GB
- Graphics: Intel(R) HD Graphics 4600
- OS Type: 64 – bits
- Operating System: Windows 7.0

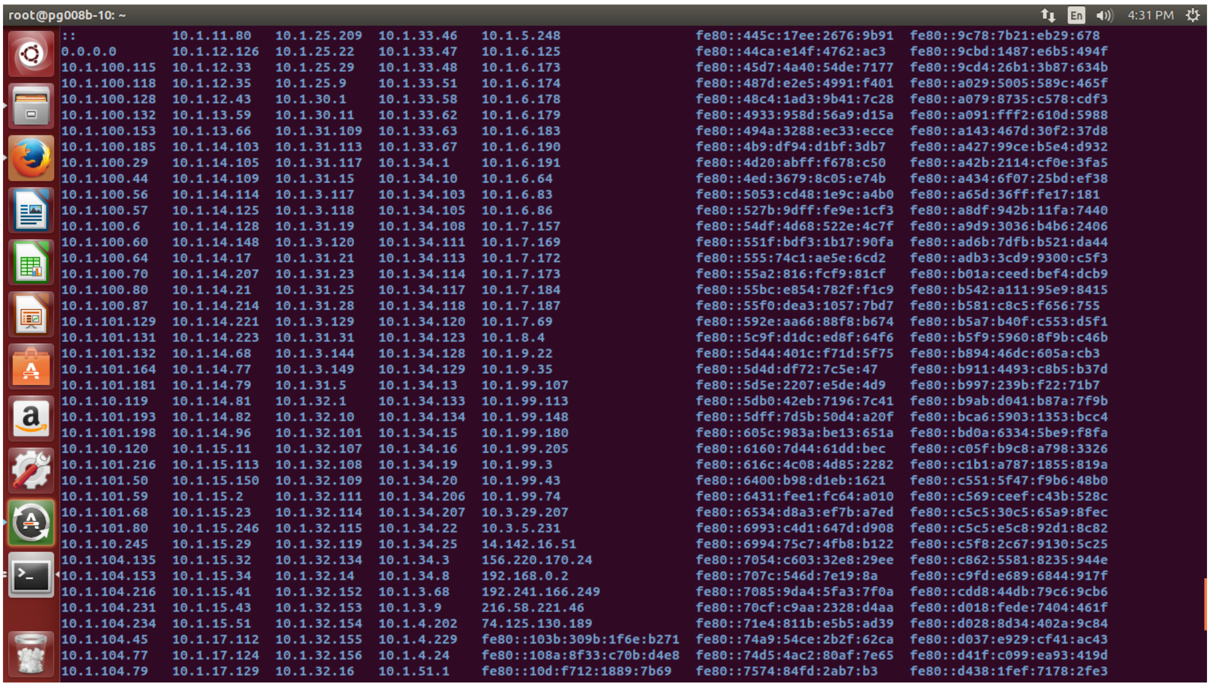
5.2 Snort Intrusion Detection System

Snort is basically use for intrusion detection system and intrusion prevention system where is NIDS(network intrusion detection system) fundamentals used in snort. Snort perform network traffic analysis where it is multi-mode network traffic packet analysis tool. Snort is open source software which is used for packet sniffing,forensic analysis, and NIDS.



```
usman@ubuntu:~$ /usr/local/bin/snort -V

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.9.0 GRE (Build 56)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2016 Cisco and/or its affiliates. All rights reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.7.4
           Using PCRE version: 8.38 2015-11-23
           Using ZLIB version: 1.2.8
```

Figure 5.1: Snort Initialization



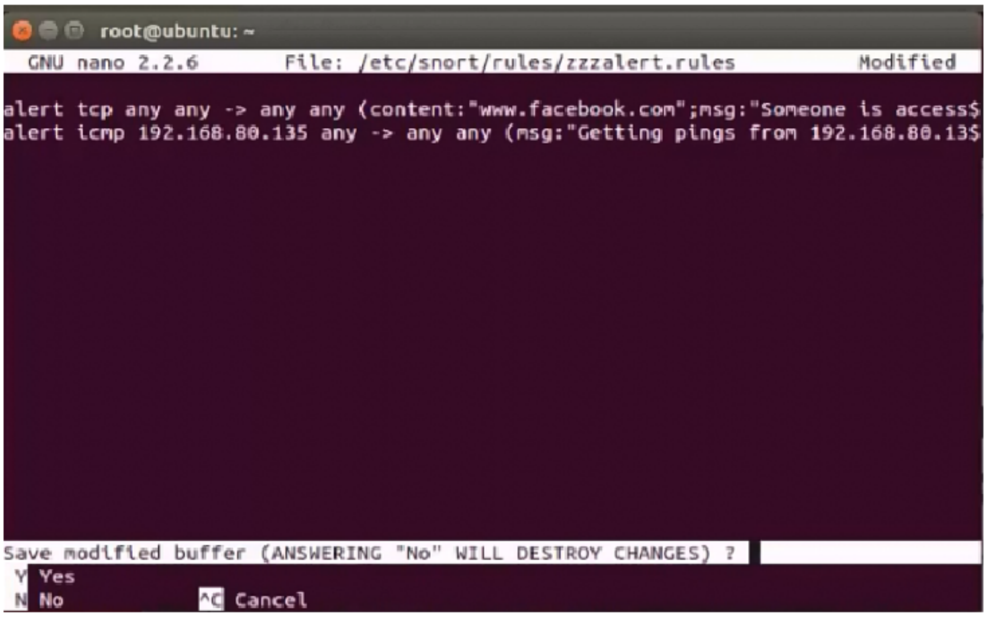
```
::            10.1.11.80   10.1.25.209  10.1.33.46   10.1.5.248                 fe80::445c:17ee:2676:9b91  fe80::9c78:7b21:eb29:678
0.0.0.0       10.1.12.126  10.1.25.22   10.1.33.47   10.1.6.125                 fe80::44ca:e14f:4762:ac3   fe80::9cbd:1487:e6b5:494f
10.1.100.115  10.1.12.33   10.1.25.29   10.1.33.48   10.1.6.173                 fe80::45d7:4a40:54de:7177  fe80::9cd4:26b1:3b87:634b
10.1.100.118  10.1.12.35   10.1.25.9    10.1.33.51   10.1.6.174                 fe80::487d:e2e5:4991:f401  fe80::a029:5005:589c:465f
10.1.100.128  10.1.12.43   10.1.30.1    10.1.33.58   10.1.6.178                 fe80::48c4:1ad3:9b41:7c28  fe80::a079:8735:c578:cdf3
10.1.100.132  10.1.13.59   10.1.30.11   10.1.33.62   10.1.6.179                 fe80::4933:958d:56a9:d15a  fe80::a091:fff2:610d:5988
10.1.100.153  10.1.13.66   10.1.31.109  10.1.33.63   10.1.6.183                 fe80::494a:3288:ec33:ecce  fe80::a143:467d:30f2:37d8
10.1.100.185  10.1.14.103  10.1.31.113  10.1.33.67   10.1.6.190                 fe80::4b9:df94:d1bf:3db7   fe80::a427:99ce:b5e4:d932
10.1.100.29   10.1.14.105  10.1.31.117  10.1.34.1    10.1.6.191                 fe80::4d20:abff:f678:c50   fe80::a42b:2114:cf0e:3fa5
10.1.100.44   10.1.14.109  10.1.31.15   10.1.34.10   10.1.6.64                  fe80::4ed:3679:8c05:e74b   fe80::a434:6f07:25bd:ef38
10.1.100.56   10.1.14.114  10.1.3.117   10.1.34.103  10.1.6.83                  fe80::5053:cd48:1e9c:a4b0  fe80::a65d:36ff:fe17:181
10.1.100.57   10.1.14.125  10.1.3.118   10.1.34.105  10.1.6.86                  fe80::527b:9dff:fe9e:1cf3  fe80::a8df:942b:11fa:7440
10.1.100.6    10.1.14.128  10.1.31.19   10.1.34.108  10.1.7.157                 fe80::54df:4d68:522e:4c7f  fe80::a9d9:3036:b4b6:2406
10.1.100.60   10.1.14.148  10.1.3.120   10.1.34.111  10.1.7.169                 fe80::551f:bdf3:1b17:90fa  fe80::ad6b:7dfb:b521:da44
10.1.100.64   10.1.14.17   10.1.31.21   10.1.34.113  10.1.7.172                 fe80::555:74c1:ae5e:6cd2   fe80::adb3:3cd9:9300:c5f3
10.1.100.70   10.1.14.207  10.1.31.23   10.1.34.114  10.1.7.173                 fe80::55a2:816:fcf9:81cf   fe80::b01a:ceed:bef4:dcb9
10.1.100.80   10.1.14.21   10.1.31.25   10.1.34.117  10.1.7.184                 fe80::55bc:e854:782f:f1c9  fe80::b542:a111:95e9:8415
10.1.100.87   10.1.14.214  10.1.31.28   10.1.34.118  10.1.7.187                 fe80::55f0:dea3:1057:7bd7  fe80::b581:c8c5:f656:755
10.1.101.129  10.1.14.221  10.1.3.129   10.1.34.120  10.1.7.69                  fe80::592e:aa66:88f8:b674  fe80::b5a7:b40f:c553:d5f1
10.1.101.131  10.1.14.223  10.1.31.31   10.1.34.123  10.1.8.4                   fe80::5c9f:d1dc:ed8f:64f6  fe80::b5f9:5960:8f9b:c46b
10.1.101.132  10.1.14.68   10.1.3.144   10.1.34.128  10.1.9.22                  fe80::5d44:401c:f71d:5f75  fe80::b894:46dc:605a:cb3
10.1.101.164  10.1.14.77   10.1.3.149   10.1.34.129  10.1.9.35                  fe80::5d4d:df72:7c5e:47    fe80::b911:4493:c8b5:b37d
10.1.101.181  10.1.14.79   10.1.31.5    10.1.34.13   10.1.99.107                fe80::5d5e:2207:e5de:4d9   fe80::b997:239b:f22:71b7
10.1.10.119   10.1.14.81   10.1.32.1    10.1.34.133  10.1.99.113                fe80::5db0:42eb:7196:7c41  fe80::b9ab:d041:b87a:7f9b
10.1.101.193  10.1.14.82   10.1.32.10   10.1.34.134  10.1.99.148                fe80::5dff:7d5b:50d4:a20f  fe80::bca6:5903:1353:bcc4
10.1.101.198  10.1.14.96   10.1.32.101  10.1.34.15   10.1.99.180                fe80::605c:983a:be13:651a  fe80::bd0a:6334:5be9:f8fa
10.1.10.120   10.1.15.11   10.1.32.107  10.1.34.16   10.1.99.205                fe80::6160:7d44:61dd:bec   fe80::c05f:b9c8:a798:3326
10.1.101.216  10.1.15.113  10.1.32.108  10.1.34.19   10.1.99.3                  fe80::616c:4c08:4d85:2282  fe80::c1b1:a787:1855:819a
10.1.101.50   10.1.15.150  10.1.32.109  10.1.34.20   10.1.99.43                 fe80::6400:b98:d1eb:1621   fe80::c551:5f47:f9b6:48b0
10.1.101.59   10.1.15.2    10.1.32.111  10.1.34.206  10.1.99.74                 fe80::6431:fee1:fc64:a010  fe80::c569:ceef:c43b:528c
10.1.101.68   10.1.15.23   10.1.32.114  10.1.34.207  10.3.29.207                fe80::6534:d8a3:ef7b:a7ed  fe80::c5c5:30c5:65a9:8fec
10.1.101.80   10.1.15.246  10.1.32.115  10.1.34.22   10.3.5.231                 fe80::6993:c4d1:647d:d908  fe80::c5c5:e5c8:92d1:8c82
10.1.10.245   10.1.15.29   10.1.32.119  10.1.34.25   14.142.16.51               fe80::6994:75c7:4fb8:b122  fe80::c5f8:2c67:9130:5c25
10.1.104.135  10.1.15.32   10.1.32.134  10.1.34.3    156.220.170.24             fe80::7054:c603:32e8:29ee  fe80::c862:5581:8235:944e
10.1.104.153  10.1.15.34   10.1.32.14   10.1.34.8    192.168.0.2                fe80::707c:546d:7e19:8a    fe80::c9fd:e689:6844:917f
10.1.104.216  10.1.15.41   10.1.32.152  10.1.3.68    192.241.166.249            fe80::7085:9da4:5fa3:7f0a  fe80::cdd8:44db:79c6:9cb6
10.1.104.231  10.1.15.43   10.1.32.153  10.1.3.9     216.58.221.46              fe80::70cf:c9aa:2328:d4aa  fe80::d018:fede:7404:461f
10.1.104.234  10.1.15.51   10.1.32.154  10.1.4.202   74.125.130.189             fe80::71e4:811b:e5b5:ad39  fe80::d028:8d34:402a:9c84
10.1.104.45   10.1.17.112  10.1.32.155  10.1.4.229   fe80::103b:309b:1f6e:b271  fe80::74a9:54ce:2b2f:62ca  fe80::d037:e929:cf41:ac43
10.1.104.77   10.1.17.124  10.1.32.156  10.1.4.24    fe80::108a:8f33:c70b:d4e8  fe80::74d5:4ac2:80af:7e65  fe80::d41f:c099:ea93:419d
10.1.104.79   10.1.17.129  10.1.32.16   10.1.51.1    fe80::10d:f712:1889:7b69   fe80::7574:84fd:2ab7:b3    fe80::d438:1fef:7178:2fe3
```

Figure 5.2: Snort Traces

- As shown in the diagram snort UI after successfully installed.mainly snort has four capabilities like Packet Sniffer ,Packet Logger ,Network Intrusion Detection, and Network Intrusion Prevention.[17]we can place snort at either VM(virtual machine)[?]. In packet sniffer mode it will read packets(based on rule) and display on console, in packet log mode log all packets into a local log, and standardized into directories by their IP add.i.e:./snort –dev –l ./log –h 192.168.1.0/24, and in NIDS mode it will scan packets by given set of rules. And its output is either in ASCII or binary format, i.e:./snort –d –l ./log –h 192.168.1.0/24 –c snort.conf.

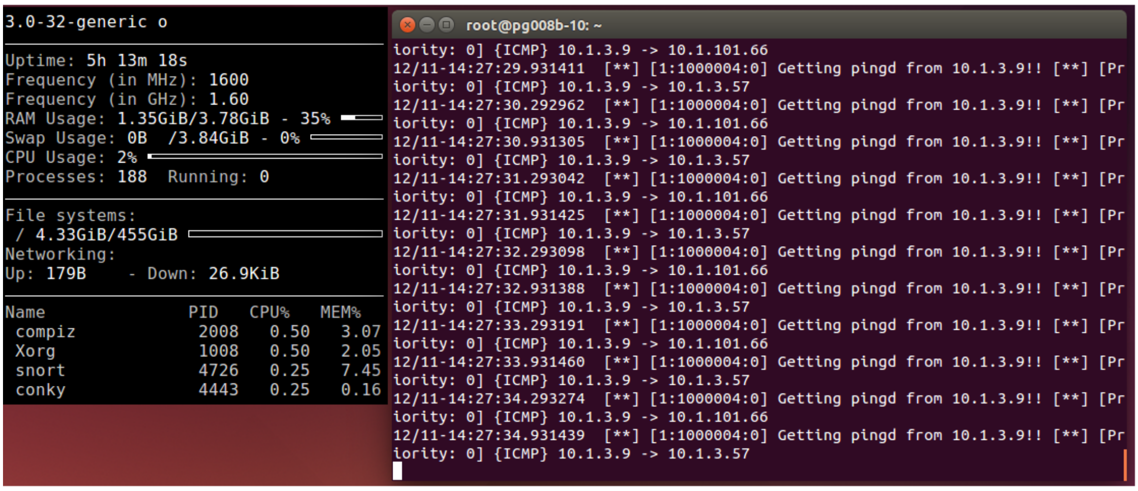
- Installing a Snort we create a rule with command and by setting a rule like we generate alert at TCP and ICMP(ping) packets are detected at the host. Another specification in this is we create DDoS detection rule also in snort

Which are as shown in figure:



```
  GNU nano 2.2.6         File: /etc/snort/rules/zzzalert.rules           Modified

alert tcp any any -> any any (content:"www.facebook.com";msg:"Someone is access$
alert icmp 192.168.80.135 any -> any any (msg:"Getting pings from 192.168.80.13$
















Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?
 Y Yes
 N No           ^C Cancel
```

Figure 5.3: Snort Rules

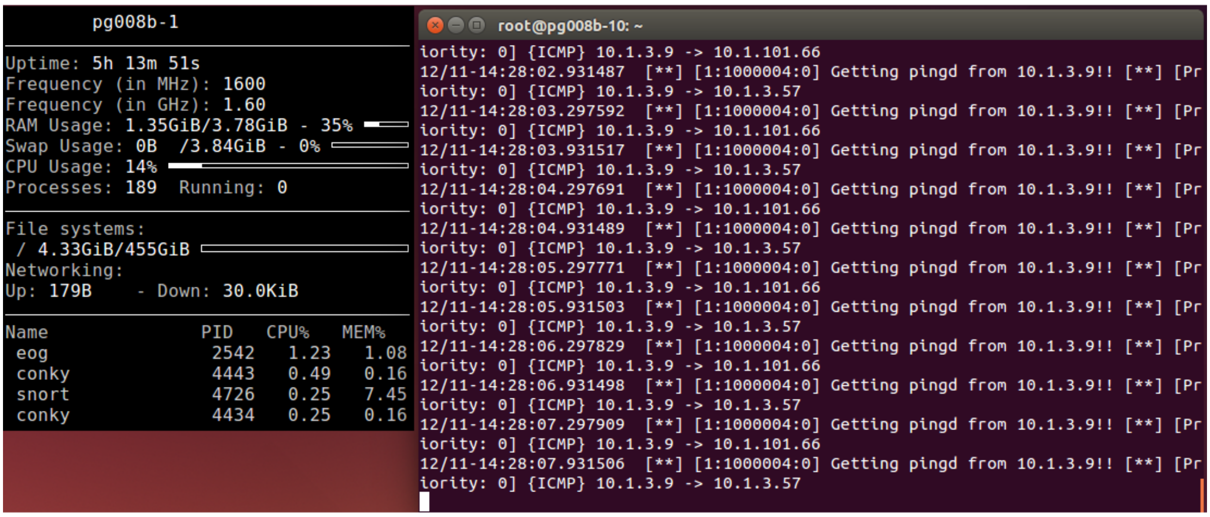


```
3.0-32-generic o

Uptime: 5h 13m 18s
Frequency (in MHz): 1600
Frequency (in GHz): 1.60
RAM Usage: 1.35GiB/3.78GiB - 35%
Swap Usage: 0B  /3.84GiB - 0%
CPU Usage: 2%
Processes: 188  Running: 0

File systems:
 / 4.33GiB/455GiB
Networking:
Up: 179B     - Down: 26.9KiB

Name                 PID    CPU%    MEM%
 compiz             2008    0.50    3.07
 Xorg               1008    0.50    2.05
 snort              4726    0.25    7.45
 conky              4443    0.25    0.16
```

```
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:27:29.931411  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
12/11-14:27:30.292962  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:27:30.931305  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
12/11-14:27:31.293042  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:27:31.931425  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
12/11-14:27:32.293098  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:27:32.931388  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
12/11-14:27:33.293191  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:27:33.931460  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
12/11-14:27:34.293274  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:27:34.931439  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
```

Figure 5.4: Traces and system utilization

But as aim of the research here when attack happened resources of cloud are affected as shown in figure like in figure percentage of CPU usage suddenly raise upto 14 precentage hence it proved that DDoS attack affect on IaaS.



```
pg008b-1

Uptime: 5h 13m 51s
Frequency (in MHz): 1600
Frequency (in GHz): 1.60
RAM Usage: 1.35GiB/3.78GiB - 35%
Swap Usage: 0B  /3.84GiB - 0%
CPU Usage: 14%
Processes: 189  Running: 0

File systems:
 / 4.33GiB/455GiB
Networking:
Up: 179B     - Down: 30.0KiB

Name                 PID   CPU%   MEM%
 eog                 2542   1.23   1.08
 conky               4443   0.49   0.16
 snort               4726   0.25   7.45
 conky               4434   0.25   0.16
```

```
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:28:02.931487  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
12/11-14:28:03.297592  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:28:03.931517  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
12/11-14:28:04.297691  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:28:04.931489  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
12/11-14:28:05.297771  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:28:05.931503  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
12/11-14:28:06.297829  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:28:06.931498  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
12/11-14:28:07.297909  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.101.66
12/11-14:28:07.931506  [**] [1:1000004:0] Getting pingd from 10.1.3.9!! [**] [Pr
iority: 0] {ICMP} 10.1.3.9 -> 10.1.3.57
```

Figure 5.5: Traces and Utilization

5.3 Results

Result of CPU and RAM case-I: In case-I shows in figure(4) represent result of CPU usage and RAM usage. As per the graph shown above it display the result for the normal memory usage and CPU utilization.

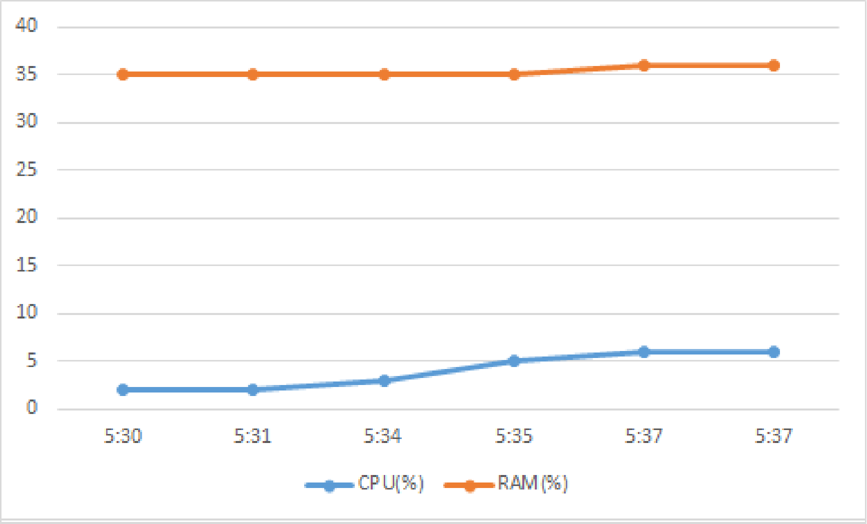


Figure 5.6: Result of CPU and RAM case-I(X axis-Time in minutes,Y axis- Utilization in Percentage)

Result of CPU and RAM case-II: In case-II shows in figure(5) represent result of CPU usage and RAM usage. As per the graph shown above it display the result for the condition in which system experiencing higher memory usage and capital CPU utilization.

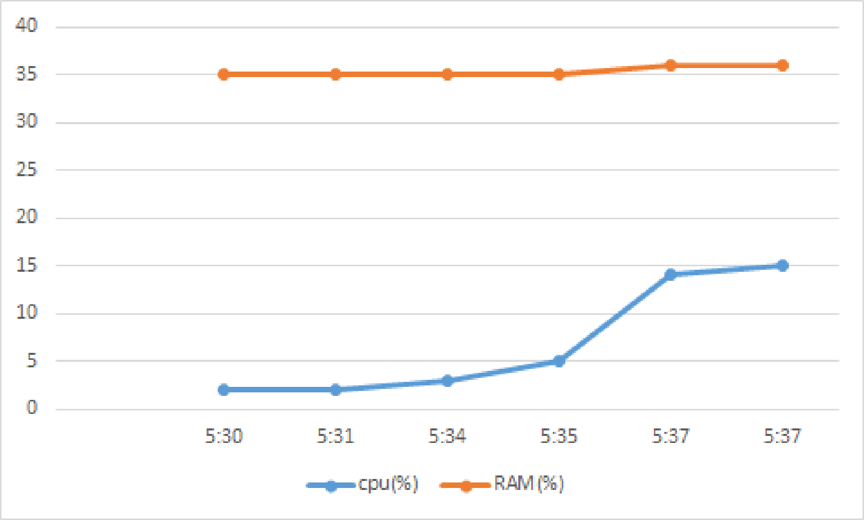


Figure 5.7: Result of CPU and RAM case-II(X axis-Time in minutes,Y axis- Utilization in Percentage)

As shown in figure-5.8 detection of highly affected VM which has less CPU and memory which are near to 0 value but the resources which has more CPU and memory shows in red color that is VM-4 it means at detection time there is VM-4,VM-9,VM-10,VM-7 and many more are not affected by attack but VM-1 and VM-8 are highly affected by DDoS attack. these all the learning process done in some number of episodes so it should be confirmed that attack has happened.

By inspection, we will be carrying our the agent in the long-term to pick out VM-4 as the more resource available, with VM-9 following second, and VM-10 following third, etc.

Taking any one of the VMs gives you a stochastic reward of either R=+1 for success, or R=0 for failure. Our objective is to agent travel the VMs one-by-one in sequence such that we maximize our total reward collected in the long run.

we approach the VMs affected by DDoS attack problem with a classical reinforcement learning technique of an epsilon-greedy agent with a learning framework of reward-average sampling to calculate the action-value Q(a) to suggest the agent improve their next action opinion for long-term reward maximization. The Python code implementation of this algorithm solution can be found.

Now to the experiments results. We execute 1000 experiments for the agent to start from scratch with epsilon analysis probability of 10%, and trained the agent for 10,000 episodes per experiment. The average proportion of VM chosen by the agent as a function

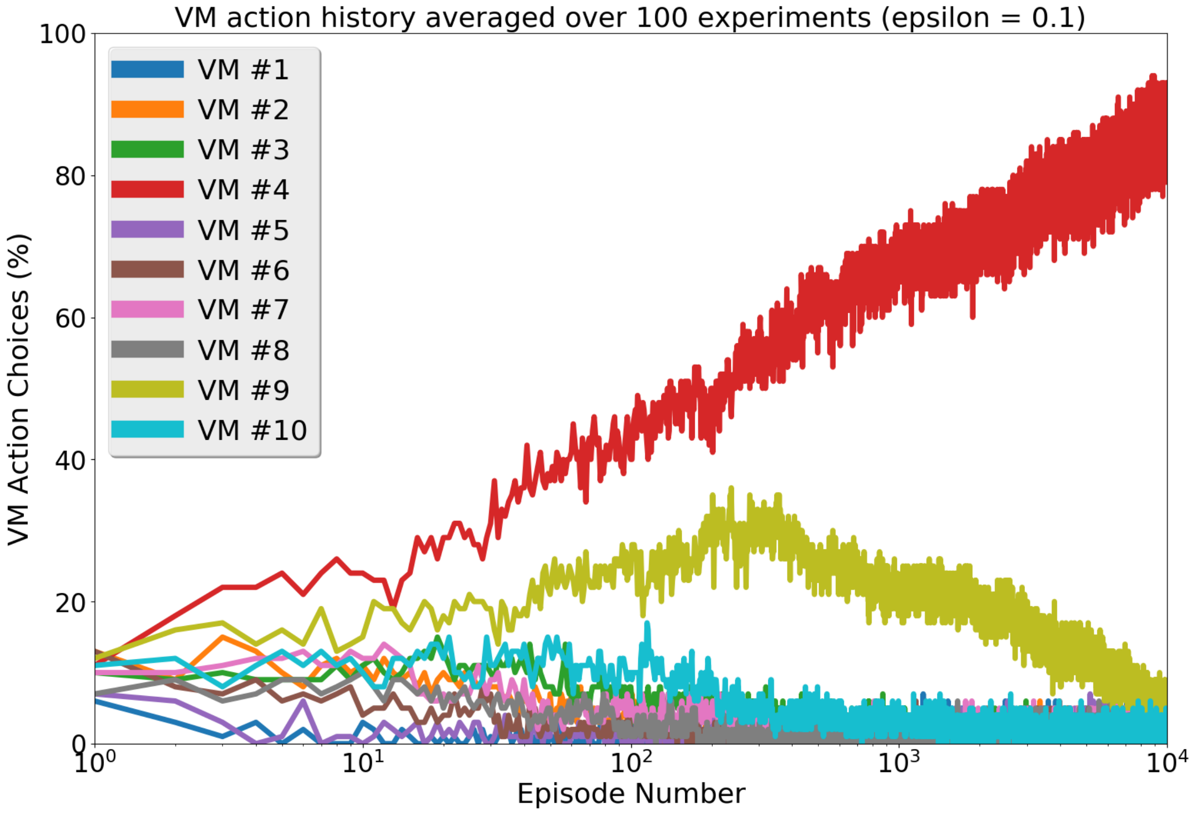


Figure 5.8: VM affected by an attack

of episode number is depicted in Fig-5.8.

According to figure-5.9 VMs those have more CPU and Memory gain Reward history over 100 experiments in same number of episodes.

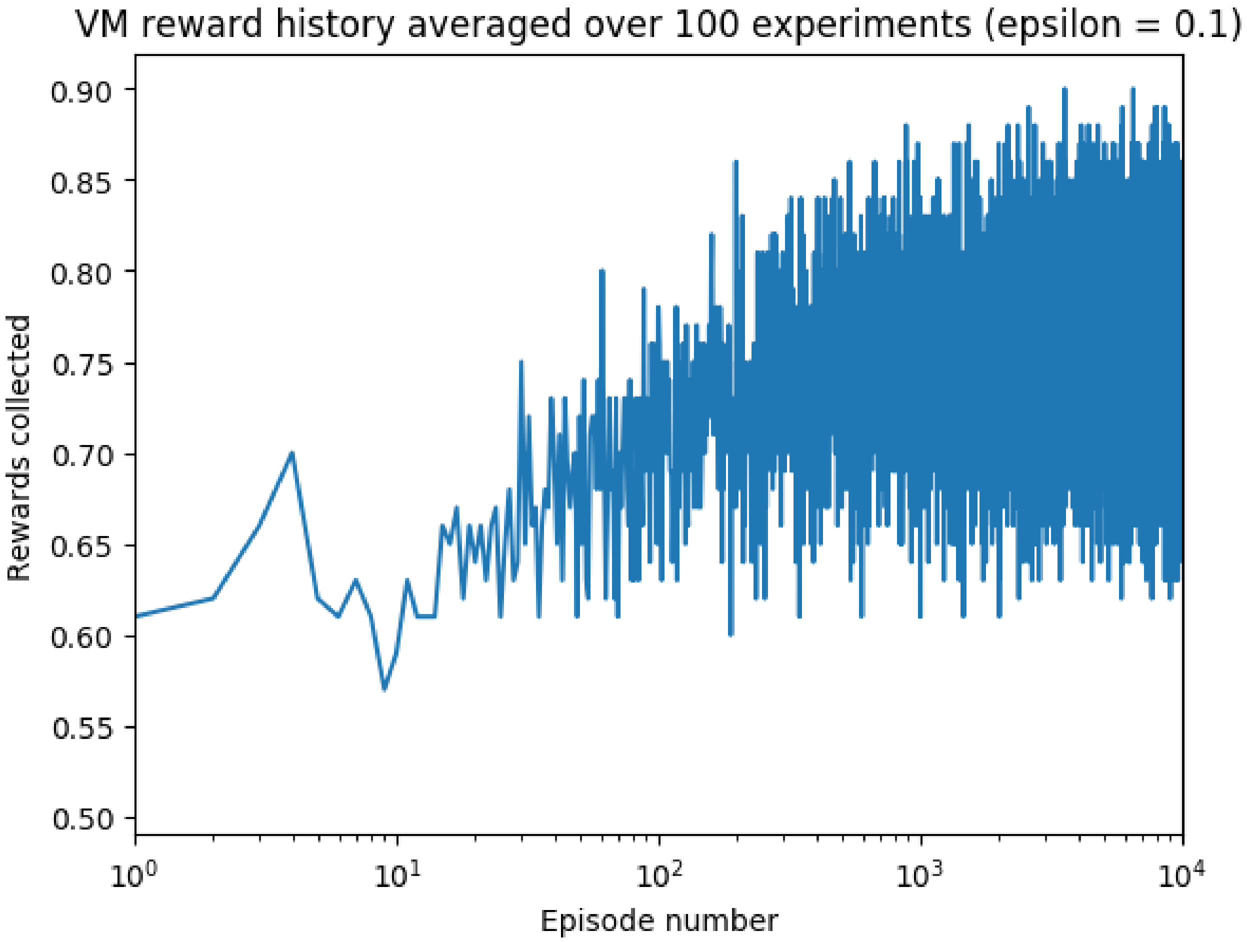


Figure 5.9: VM Reward History

As shown in figure-5.12, there is 7 nodes which are plotted randomly where some of the nodes are legitimate users and some of are attackers. before the learning process it were plotted anywhere into plan.

As shown in figure-5.13, Using Q-Learning method gives a positive or negative Reward basis on policy. where policy defined by ping request coming from the node. suppose, node gives more than 100 ping request which means it would be an attacker. Otherwise it would be a legitimate User. According to this scenario Reinforcement learning gives negative reward to the attackers and gives positive reward to the legitimate users.

Afterward, we can stop or prevent the attackers to not to do nefarious activities or if we can't stop attackers in specific time then we will go for load balancing technique to handled users request.

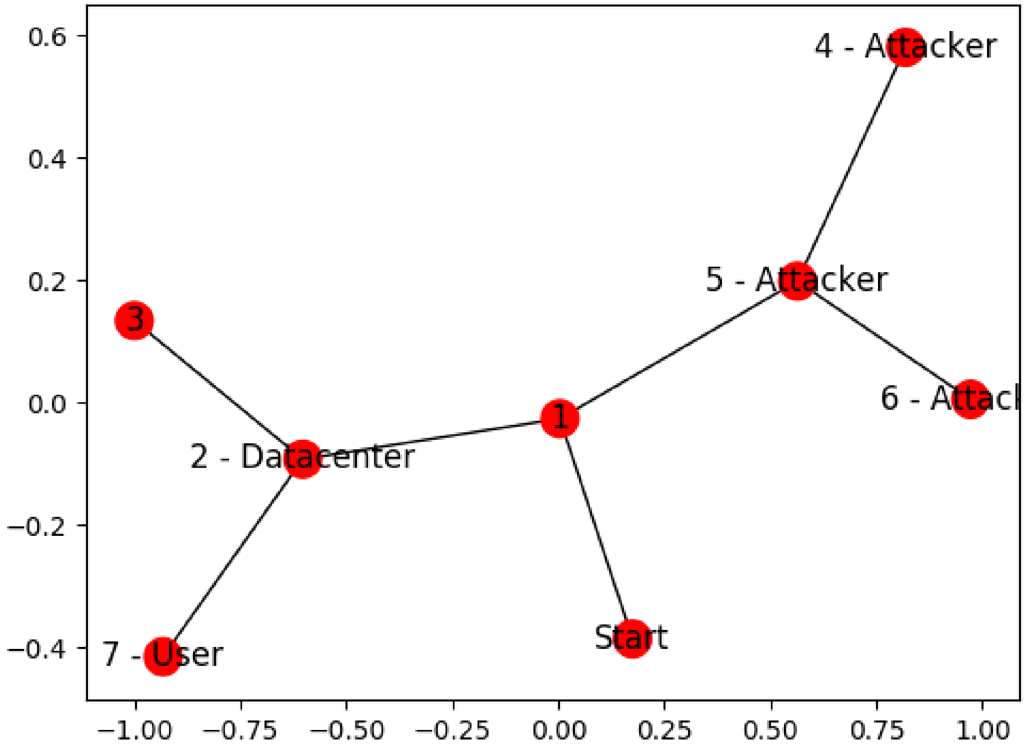


Figure 5.10: X and Y axis- Positive and Negative Reward

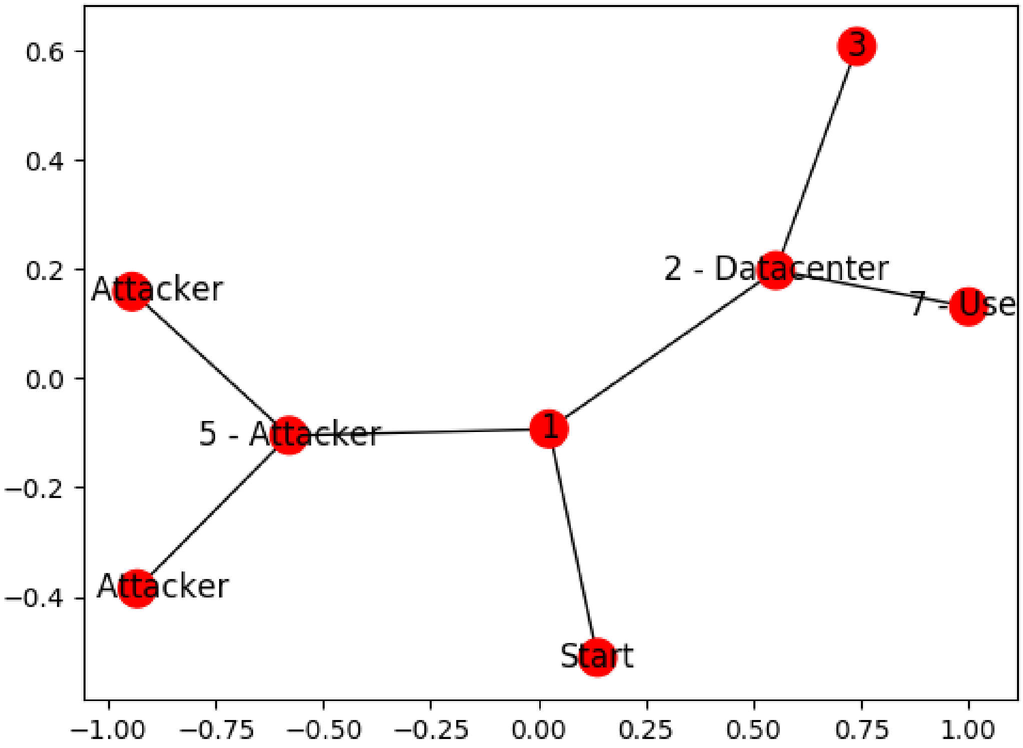


Figure 5.11: Trained Model by RL,X and Y axis- Positive and Negative Reward

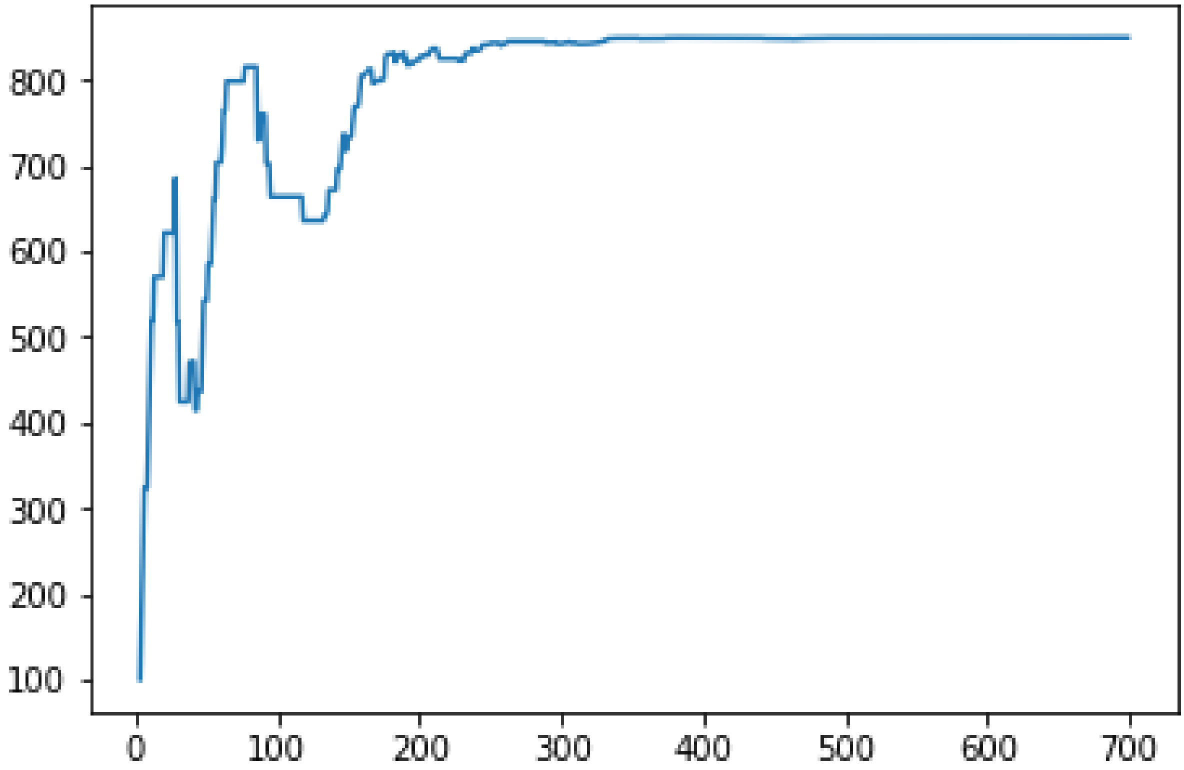


Figure 5.12: Trained Model by RL,X and Y axis- Positive and Negative Reward

Chapter 6

Conclusion and Future Work

According to the report of IBM Cloud Managed Service Compliance and Audit, we can say that Cloud is getting more vulnerable day by day. The DDoS attack is one of the most vulnerable attacks.DDOS attack detection is a very complex and complicated problem for cloud computing technology. For detection of attack at IaaS level, we are using Reinforcement Algorithm in which we are using network packet which is captured by implementing IDS to learn and train system to detect the attack at the initial level. Using RL method we use Q-Learning technique for selection of optimal policy and defined policy by value function. Using Q-Learning method it will give a Positive or Negative Reward based on policy. Here if an attacker found it will give a Negative Reward otherwise gives a Positive Reward for a legitimate Users. By use of RL, we create an Agent which perform an Action for a particular State and maximize the reward based on reward history.in future work, we will be implementing Hierarchical Reinforcement Learning(HRL) for the various supporting policies to solve the main task of the system. The benefits of using HRL will be, to use policies which can act as a backup source to deal with a situation where the primary policy will fail.

Bibliography

[1] B. Hayes, "Cloud computing," *Communications of the ACM*, vol. 51, no. 7, pp. 9–11, 2008.

[2] P. Mell, T. Grance, *et al.*, "The nist definition of cloud computing," 2011.

[3] Y. Jadeja and K. Modi, "Cloud computing-concepts, architecture and challenges," in *Computing, Electronics and Electrical Technologies (ICCEET), 2012 International Conference on*, pp. 877–880, IEEE, 2012.

[4] R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, and J. Molina, "Controlling data in the cloud: outsourcing computation without outsourcing control," in *Proceedings of the 2009 ACM workshop on Cloud computing security*, pp. 85–90, ACM, 2009.

[5] A. Beloglazov and R. Buyya, "Energy efficient resource management in virtualized cloud data centers," in *Proceedings of the 2010 10th IEEE/ACM international conference on cluster, cloud and grid computing*, pp. 826–831, IEEE Computer Society, 2010.

[6] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, *et al.*, "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[7] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," *Journal of internet services and applications*, vol. 1, no. 1, pp. 7–18, 2010.

[8] K. Ren, C. Wang, and Q. Wang, "Security challenges for the public cloud," *IEEE Internet Computing*, vol. 16, no. 1, pp. 69–73, 2012.

[9] M. Ali, S. U. Khan, and A. V. Vasilakos, "Security in cloud computing: Opportunities and challenges," *Information Sciences*, vol. 305, pp. 357–383, 2015.

[10] L. Wang, R. Ranjan, J. Chen, and B. Benatallah, *Cloud computing: methodology, systems, and applications.* CRC Press, 2017.

[11] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A break in the clouds: towards a cloud definition," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 1, pp. 50–55, 2008.

[12] M.-G. Avram, "Advantages and challenges of adopting cloud computing from an enterprise perspective," *Procedia Technology*, vol. 12, pp. 529–534, 2014.

[13] A. M. Lonea, D. E. Popescu, and H. Tianfield, "Detecting ddos attacks in cloud computing environment," *International Journal of Computers Communications & Control*, vol. 8, no. 1, pp. 70–78, 2013.

[14] M. T. Khorshed, A. S. Ali, and S. A. Wasimi, "A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing," *Future Generation computer systems*, vol. 28, no. 6, pp. 833–851, 2012.

[15] M. Cusumano, "Cloud computing and saas as new computing platforms," *Communications of the ACM*, vol. 53, no. 4, pp. 27–29, 2010.

[16] N. G. Kejriwal and P. Judge, "Method for detecting malicious javascript," July 22 2014. US Patent 8,789,178.

[17] A. M. Lonea, D. E. Popescu, and H. Tianfield, "Detecting ddos attacks in cloud computing environment," *International Journal of Computers Communications & Control*, vol. 8, no. 1, pp. 70–78, 2013.

[18] G. Sanchika, K. Padam, and A. Ajith, "A profile based network intrusion detection and prevention system for securing cloud environment," *International Journal of Distributed Sensor Networks, Hindawi Publishing Corporation*, vol. 2013.

[19] B. Singh and S. Panda, "An adaptive approach to mitigate ddos attacks in cloud," *IJACSA*, vol. 6, no. 10, pp. 47–52, 2015.

[20] B. K. Devi and T. Subbulakshmi, “A comparative analysis of security methods for ddos attacks in the cloud computing environment,” *Indian Journal of Science and Technology*, vol. 9, no. 34, 2016.

[21] M. Ali, S. U. Khan, and A. V. Vasilakos, “Security in cloud computing: Opportunities and challenges,” *Information Sciences*, vol. 305, pp. 357–383, 2015.

[22] M. Masdari and M. Jalali, “A survey and taxonomy of dos attacks in cloud computing,” *Security and Communication Networks*, vol. 9, no. 16, pp. 3724–3751, 2016.

[23] O. Osanaiye, K.-K. R. Choo, and M. Dlodlo, “Distributed denial of service (ddos) resilience in cloud: review and conceptual cloud ddos mitigation framework,” *Journal of Network and Computer Applications*, vol. 67, pp. 147–165, 2016.

[24] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and R. Buyya, “Ddos attacks in cloud computing: issues, taxonomy, and future directions,” *Computer Communications*, 2017.

[25] U. T. M. F. to Achieve Effective Security Mechanisms in Cloud Environment., “2017,” *Hicham Toumi,Bouchra Marzak,Amal Talea,Ahmed Eddaoui and Mohamed Talea*, vol. 4, no. 3, 59-64.

[26] Z. Shen, L. Li, F. Yan, and X. Wu, “Cloud computing system based on trusted computing platform,” in *Intelligent Computation Technology and Automation (ICICTA), 2010 International Conference on*, vol. 1, pp. 942–945, IEEE, 2010.

[27] Z. Shen and Q. Tong, “The security of cloud computing system enabled by trusted computing technology,” in *Signal Processing Systems (ICSPS), 2010 2nd International Conference on*, vol. 2, pp. V2–11, IEEE, 2010.

[28] X.-Y. Li, L.-T. Zhou, Y. Shi, and Y. Guo, “A trusted computing environment model in cloud architecture,” in *Machine Learning and Cybernetics (ICMLC), 2010 International Conference on*, vol. 6, pp. 2843–2848, IEEE, 2010.

[29] H. Takabi, J. B. Joshi, and G.-J. Ahn, “Securecloud: Towards a comprehensive security framework for cloud computing environments,” in *Computer Software and Applications Conference Workshops (COMPSACW), 2010 IEEE 34th Annual*, pp. 393–398, IEEE, 2010.

[30] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities," in *High Performance Computing and Communications, 2008. HPCC'08. 10th IEEE International Conference on*, pp. 5–13, Ieee, 2008.

[31] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless communications and mobile computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[32] S. Bhardwaj, L. Jain, and S. Jain, "Cloud computing: A study of infrastructure as a service (iaas)," *International Journal of engineering and information Technology*, vol. 2, no. 1, pp. 60–63, 2010.

[33] R. L. Krutz and R. D. Vines, *Cloud security: A comprehensive guide to secure cloud computing.* Wiley Publishing, 2010.

[34] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *Journal of network and computer applications*, vol. 34, no. 1, pp. 1–11, 2011.

[35] D. KS and A. Kamath, "Survey on techniques of data mining and its applications," 2017.

[36] P. Mishra, E. S. Pilli, V. Varadharajan, and U. Tupakula, "Intrusion detection techniques in cloud environment: A survey," *Journal of Network and Computer Applications*, vol. 77, pp. 18–47, 2017.

[37] P. Pandey, D. Pandey, and S. Kumar, "Reinforcement learning by comparing immediate reward," *arXiv preprint arXiv:1009.2566*, 2010.

[38] X. Xu, Y. Sun, and Z. Huang, "Defending ddos attacks using hidden markov models and cooperative reinforcement learning," *Intelligence and Security Informatics*, pp. 196–207, 2007.

[39] A. Patel, M. Taghavi, K. Bakhtiyari, and J. C. JúNior, "An intrusion detection and prevention system in cloud computing: A systematic review," *Journal of network and computer applications*, vol. 36, no. 1, pp. 25–41, 2013.