REINFORCEMENT LEARNING TO REDUCE THE ATTACK SURFACE IN SELF SERVICE CLOUD COMPUTING

By

BALAJI GANESULA

Bachelor of Technology in Computer Science

And Engineering

SRM University

Chennai, TN, India

2010

Submitted to the Faculty of the Graduate College of the

Oklahoma State University in partial fulfillment of

the requirements for the Degree of

MASTER OF SCIENCE

December 2013

REINFORCEMENT LEARNING TO REDUCE

THEATTACK SURFACE

IN SELF SERVICE CLOUD COMPUTING

Thesis Approved:

Dr. Johnson Thomas

Thesis Adviser

Dr. David Cline

Dr. Tingting Chen

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, Dr. Jonson Thomas for his excellent guidance, patience, and providing me with an excellent atmosphere for doing research. His guidance helped me to successfully complete my research.

Besides my advisor, I would like to thank rest of the thesis committee: Dr.David Cline, Dr.Ting Ting Chen for their encouragement and insightful comments.

I thank my fellow classmate P.Praveen Kumar for his suggestions and support.

Last but not the least; I would like to thank my family for supporting me throughout my life.

Name: BALAJI GANESULA

Date of Degree: DECEMBER 2013

Title of Study: REINFORCEMENT LEARNING TO REDUCE THE ATTACK SURFACE IN SELF SERVICE CLOUD COMPUTING

Major Field: COMPUTER SCIENCE

ABSTRACT

Cloud computing offers various services which are analogous to traditional data centers. The on demand supply of resources make this model of utility computing as the platform for many web based services. However security is always a major concern. This thesis proposes a new architecture called Self-service cloud computing with virtual shield (VS) to secure the entire cloud environment. Virtual shield (VS) is designed with the reinforcement learning mechanism to dynamically change the configurations of the client virtual machines (VM) in case of an attack to achieve the required security. This work introduces a novel way to measure the security of the system based on attack surface. The configurations scores generated during the learning process determines the activity of the client. The dynamic configuration of virtual machines in-case of an attack, reduces the attack surface and secures the cloud VM's.

Keywords- Self-service cloud computing; attack surface; Reinforcement learning; Virtual shield

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

CHAPTER I

INTRODUCTION

1.1 OVERVIEW

Cloud computing is a novel architecture in the field of information technology. Cloud computing offers various services which are analogous to traditional data centers. Software as a service (SAAS), Infrastructure as a service (IAAS), Application as a service (AAAS), Platform as a service (PAAS) promotes cloud computing to various organizations [7]. Cloud computing provides location independent services to the user. Resource allocation, data management, load balancing is under the control of cloud service providers. However, security is always a major concern in cyber cloud technology. The principles, methodologies, and tools for secure cloud computing are yet to be developed. Various cloud security systems such as advanced cloud systems (ASP) through secure virtualization [8], cloud protector through cloud trace back mechanism [10], hierarchical attribute encryption [11] have been proposed to enhance security in the cloud environment. However, these mechanisms degrade the performance of the system and counter only known attacks.

A novel architecture called self-service cloud computing [1] has been introduced to resolve most of the issues faced by guest virtual machines. However this system is less concerned about the security of the host operating system since most of the privileges exists within the guest Meta domain created. Moreover there is no standard way of measuring the cloud system with respect to security. This work introduces a way to measure the security of the system through an attack surface. This paper also introduces an extension to self-service cloud computing [1] to dynamically configure the privileges between the host operating system and guest virtual machines in SSC (Self Service Cloud Computing).This protects the host operating system from the guest virtual machines.

1.2 SELF SERVICE CLOUD WITH VIRTUAL SHIELD

Self-service cloud computing is a service oriented architecture which mitigated security and privacy issues related to client virtual machines. Inflexible control, which requires the cloud providers to define the security measurements like VMware introspection, migration and check pointing are handed over to client's Meta domain in the SSC. The Hypervisor and hardware are assumed to be a TRUSTED COMPUTING BASE since they are provided by trusted organizations in this architecture. In self-service cloud computing the host operating system has no privileges to view the guests virtual CPU, memory or the configuration parameters of the Meta domain. The operating system acts to initiate the boot up process and hold the privileges to shut down the virtual machines. Though this architecture provides a MUTUALLY TRUSTED SERVICE DOMAINS (MTSDs) which is a regulatory compliance between cloud providers and users it is not sufficient to handle the misuse of the cloud infrastructure by the guest operating systems.

This research focuses on shifting the privileges between the host operating system and guest virtual machines. A Virtual Shield (VS) is introduced to act according to the information provided by the MTSDs. The Virtual shield is a virtual machine designed to dynamically configure the guest virtual machines with the help of a reinforcement learning algorithm.

1.3 ATTACK SURFACE IN CLOUD COMPUTING

In cloud computing there is no standard way to measure the security of the system. The Attack surface is one way to measure the security of the system. A system with a larger attack surface is more vulnerable to attacks and vice versa. Secure cloud computing needs a practical software metrics and measurements [2]. The Attack surface in cloud computing can be categorized into 6 different categories based on the systems involved in the communication [2]. Our work introduces an IO automation model for the attack surface of self-service cloud computing with reinforcement learning and compares this model with a self-service attack surface.

1.4 REINFORCEMENT LEARNING

Reinforcement learning method is used to make the virtual shield introduced in the new architecture to learn from the environment. This learning in turn facilitates the configuration of the host and guest virtual machines dynamically. The shifting of privileges reduces the attack surface in self-service cloud computing.

1.5 PROBLEM STATEMENT

Guest operating systems misuse the cloud infrastructure for malicious activities. At the moment, there is no way to identify the attacks because most of the privileges exist within the Meta domain of the guest. This increases the attack surface and results in various attacks on the host virtual machine and hypervisor.

1.6 RESEARCH OBJECTIVE

The main objective is to reduce the attack surface of self-surface cloud computing by introducing a virtual shield in the system. The virtual shield has the capability to learn from the environment and dynamically configure the host and guest operating systems. This learning is based on the reinforcement learning methodology.

1.7 OUTLINE

The rest of the thesis is organized as follows: Chapter 2 provides the review of literature, Chapter 3 presents the proposed work, and Chapter 4 covers the simulations and results. Finally Chapter 5 concludes our paper and provides some insights into future work.

CHAPTER II

LITERATURE REVIEW

Cloud computing refers to computing over the internet where dynamically scaled shared resources (mostly virtual) are provided as a service by using virtualization platforms. Cloud Architecture is based on virtualization of the resources. Below is the basic cloud architecture [7].

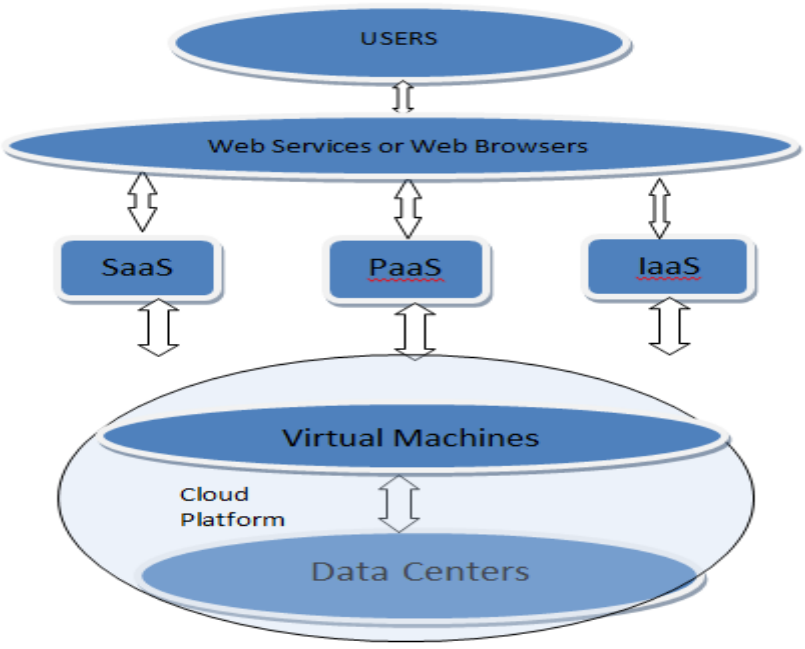

Figure 2.1 Basic Cloud Frameworks

Cloud computing utilizes a service oriented architecture to utilize the services of cloud. There are different types of virtualizations that are used in cloud computing such as Storage, Network and server virtualization [8] which yields a different set of security concerns for each type of virtualization technique used.

2.1 SECURITY ISSUES IN CLOUD COMPUTING

The client or user is unaware of which physical system the process is actually running on and where the data is stored. If a malicious user is from same Physical system he can get the data from the physical system. This is because VM's (Virtual Machines) map the data on storage provided logically but all the data resides physically on single storage. Data centers are located across the globe. User should be able to define where his data and process should reside because each country has different security policies.

We need to have a different security levels in the cloud architecture which reduces the amount of risks because of the above problems and it should be cost effective. Different data centers have different security policies and different VM's run on different zones of security leading to loss of policies and increased security concerns. Because all these VM's from different security zones communicate with each other on a Virtual Network. Any weak link will pose a severe threat to whole application [9].

Since the cloud uses virtualization, it needs to keep up to date with the latest patches for all the virtual machines which is very difficult to manage. Security configuration management is a serious problem and administrators have to keep track of each VM and all the security policies related to data and localization. The Cloud uses different service models such as Saas, Paas, Daas, Iaas, and NaaS [7], which introduces different levels of security systems for each kind of service models.

Virtualization introduces many more problems into the cloud [8]. Using Virtualization we introduce many new OS types over which the applications are run. These new OS's are a security concern. Different virtual OS's have different security mechanisms. If one of the new OS is attacked then the attacker will try to get access to the underlying physical host. This means it will affect the security of all other virtual machines running on this physical host.

The VM's communicate with each other over the network which opens avenues for the Guest to guest attack where one virtual machine tries to attack the other virtual machines. Moreover it is difficult to keep track of the VM's. In this scenario two VM's communicate with each other over a network. VM1 can get information regarding VM2 by sending queries while communicating. VM1 might be an intruder or a malicious user. Since it's difficult to keep track of VM's it's difficult to determine who the malicious user is and what information has been compromised.

Hyper jacking is one other attack where a malicious user will run a thin Hypervisor (a rootkit) on the physical system by which he will get root level access of the whole Operating system. Example is a Blue pill [9]. This lies between the Host OS and VM's. It allocates resources to the Guest OS. Regular security measures are ineffective with this kind of attack because the original OS will be compromised with this. Security between the Host and VM is another important area which introduces a threat called VM escape [9]. There are different zones in the cloud and even if a single VM is attacked it poses a threat to the whole system. Using any of these attacking techniques and other possible ways of attacking, if a malicious user can get access to a VM then he can directly communicate with Hypervisor and can get complete access to all the VM's [9].

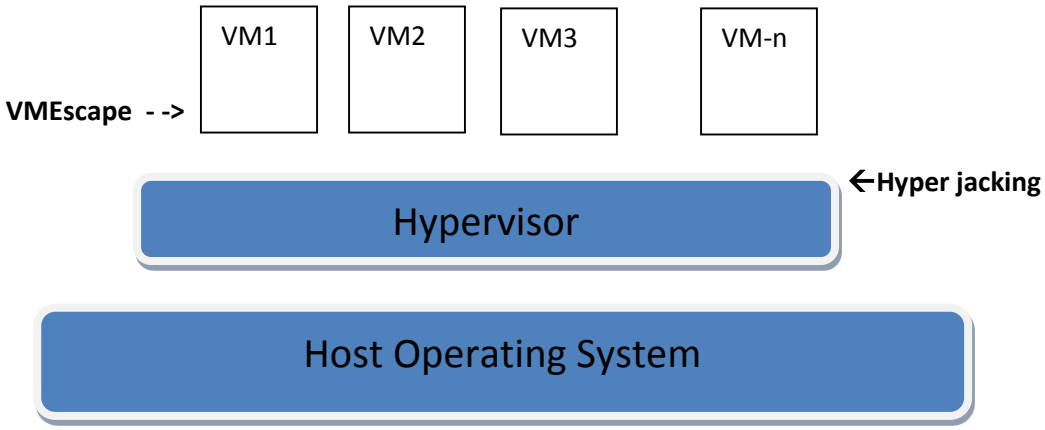

Figure 2.2 Attacks on Hypervisor and Virtual Machines

In order to avoid VM escape all the VM's should be well identified and managed. Different types of cloud are the most important targets where there will be many security concerns [9]. These different types of clouds include private clouds, public clouds and hybrid which pose a serious threat when VM's from these individual clouds communicate with each other [4]. A Private cloud by itself might be more secure but when it joins with the public cloud to form a Hybrid cloud there are huge security risks.

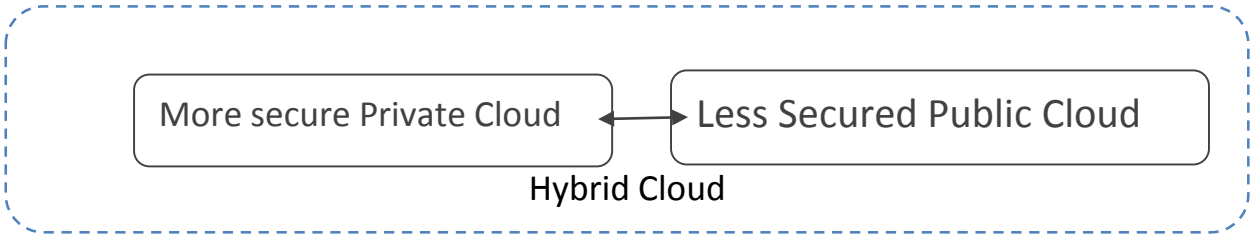

Figure 2.3 Hybrid Cloud

The virtual network should be protected along with the physical network because both of them are independent on their own [8]. Virtual network traffic is highly invisible due to virtualization. This may lead to attacks on the physical network and vice-versa.

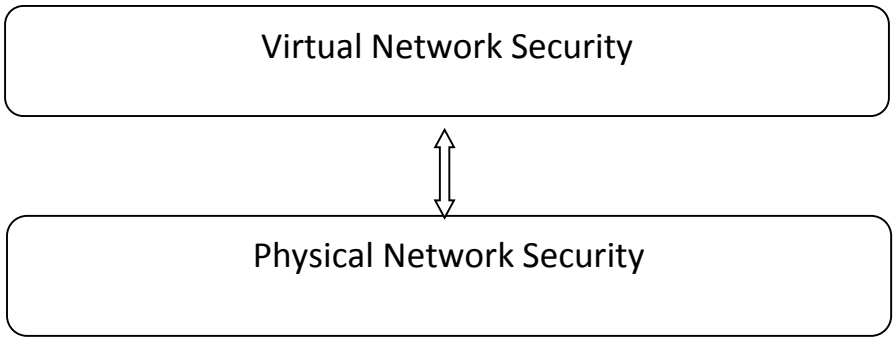

Figure 2.4 Virtual Network and Private Network

Although the network of VM's and actual physical machines are independent, any security threat to either of these will result in security concerns which may be data loss, resource wastage, flooding and loss of data integrity.

All the services (Saas, Paas, IaaS, Security as a Service, DaaS, NaaS) [7] are offered using web services or Web browsers. Hence VM security alone is not enough. Using these web services users will get access to the VM's on which these services run. So we need to secure the way users communicate with these VMs. We need to have trust when a user uses an application developed on the cloud. A different user in another VM from the cloud can communicate with the application running on the Cloud. Hence we need to have trust between the users in the cloud when they communicate with each other.

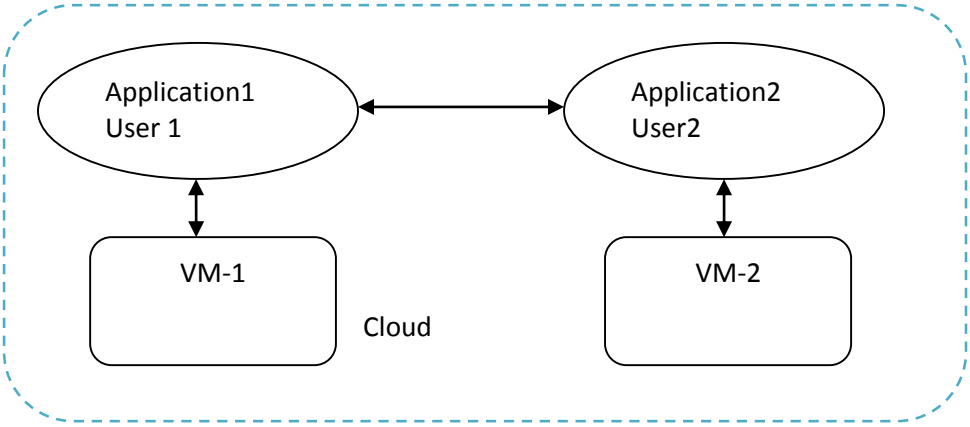

Figure 2.5 Trusts between Users

Interactive virtualization relates risks say communication between the virtualized network and virtualized server. Any security concerns will multiply the number of threats. Furthermore, VM deployment and VM data backup needs to be more secure.

The dynamic and elastic nature of cloud introduces new threat [7]. When new resources are added to the existing cloud they must be compatible with existing security policies before use. This introduces dynamic security assignment before use.

The proposed work is related to self-service cloud computing [1]. The self-service cloud computing architecture is modified to enhance the security and reduce the attack surface of the system. Attack surface is used as a measurement to compare different cloud architectures. To enhance the security of the cloud system, reinforcement learning methods can be used. Furthermore the virtual shield can be configured with different security metrics to defend against various attacks on the host virtual machine.

2.2 SELF SERVICE CLOUD COMPUTING

Self-service cloud computing is a computing model that that resolves two shortcomings in the traditional cloud architecture. Virtualization is the key to any cloud architecture [7]. Virtual machine monitors are used in many cloud architectures to administer and execute client virtual machines. These virtual machine monitors comprises of a Hypervisor, Hardware and a host virtual machine called dom0. The hypervisor and hardware are assumed to be a trusted computing base whereas the dom0 is considered to be the source of different attacks. Since most of the privileges lies within dom0 there is a high risk of utilizing this administrative domain for malicious activities.

The two major problems in the traditional cloud computing are

- Security and privacy of the client virtual machine

  The state of the client virtual machine can be inspected by dom0. It holds the privileges to inspect the contents of the client VMs and their configurations. The client virtual machines security and privacy can be compromised due to various attacks by the host virtual machine's (Dom0). Misconfiguration and malicious system administrators can be the source of attacks.

- Inflexible control over the client VMs

  Virtualization facilitates different services to the client. It has the potential to enable services like, migration, check pointing and VM introspection [1]. However the deployment of these services in the present cloud architecture is under the control of cloud infrastructure providers. The client virtual machines have no control over the adoption of these services. Upon the request of the Client, the virtual machines are

configured with these services. However, these services won't fit for all the clients. A few client VMs may use encryption to securely transfer data packets, but the service that checks the malicious content using signatures may not be able to use the encryption mechanism. The client virtual machines may need different security mechanisms for different kinds of attacks. Thus the present cloud architecture has inflexible control over the client VMs.

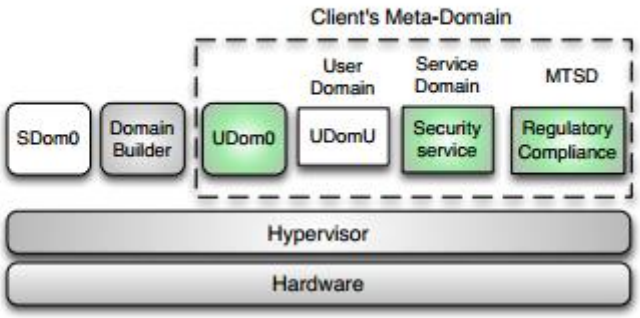

Figure 2.6 Self Service Cloud Computing

Self-service cloud computing addresses these two shortcomings by assigning more privileges to client virtual machines. The protocol is designed to protect the client virtual machines from malicious system administrators and to provide control of the services required by the client. The SSC (Self-service cloud computing) divides the entire system into two TCBs (Trusted COMPUTING BASE). The system consists of the system level TCB, with the hardware, the SSC hypervisor, the domain builder and a client-level TCB, with the Udom0 and service domains.

UDom0 is the client side per user administrative domain that can monitor and control the set of VMs of a particular client. This virtual machine attempts to start a VM in SSC. It also has the privileges to perform system services on the client virtual machines.

UDomUs are the actual client side virtual machines with the guest operating systems.

SDs (Service Domains) can be configured with required security services in the system.

MTSDs (Mutually trusted service domains) act as a regulatory compliance between cloud providers and clients. This holds the policies and mechanisms that the provider will use to control the clients VMs. The information provided by the MTSDs is the key source for the virtual shield in our prototype model.

All these comprise to form the client side Meta domain.

DomB (Domain Builder) is a virtual machine provided by the cloud provider to build the guest virtual machines upon the request from client.

SDom0 (System side administrative domain) administers the client virtual machines. It takes care of starting and stopping of the client VMs.

2.3 ATTACK SURFACE AND TAXONOMY OF ATTACKS IN CLOUD

There is a necessity to categorize the attacks in the cloud [2]. The classification of these attacks to different taxonomies helps to handle various security issues. Furthermore this classification criterion designs the attack surface for different levels in the cloud architecture.

Cloud computing can be modeled with three different types of participants, service users, service instances (Virtual Machines), and the cloud provider [3]. The interactions in the cloud will be

between any of these two classes. Similarly the attacks in the cloud can be identified into a set of interactions with in these 3 classes (service users, service instances, and cloud providers).

The six different kinds of attack surfaces have been identified in the cloud [2]:

1. service-to-user

2. user-to-service

3 cloud-to-services

4. service-to-cloud

5. cloud-to-user

6. user-to-cloud

These attack surfaces act as a measurement of security in cloud computing. The system is said to be more secure if it has a smaller attack surface.

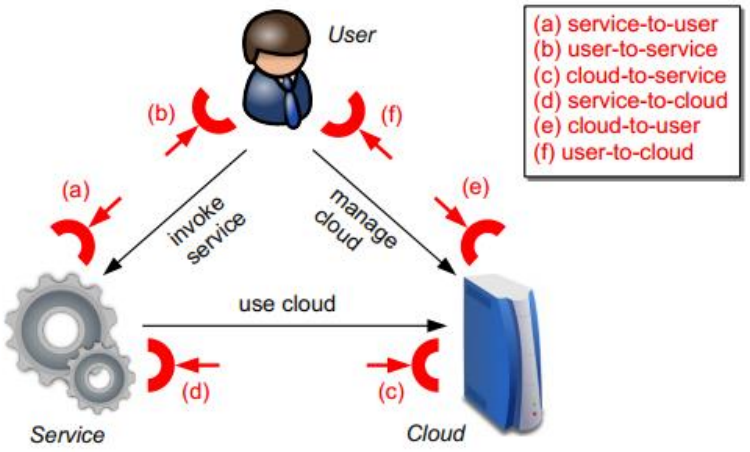

Figure 2.7 Attack Surfaces in Cloud

An IO automation model can be used to design the attack surface of the cloud [5]. Attack surface reduction and code quality improvement are complementary approaches for reducing security risk. A smaller attack surface reduces the risk of vulnerabilities by making it harder to exploit and thus lowers the potential of exploitation.

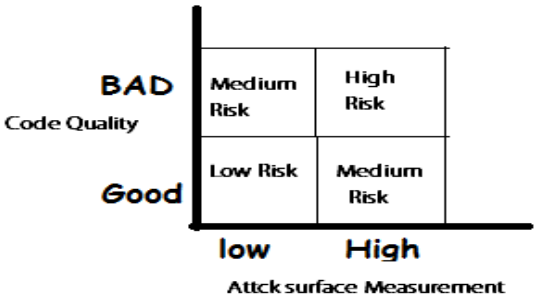

Figure 2.8 Attack Surface Measurements

2.3.1 ATTACK SURFACE METRICS

Many attacks like exploiting a buffer overflow, symlink attacks, on a system take place because the system sends data into the environment. In these attacks the attacker system and the attacked system were connected using some channels (e.g. sockets), invoking system methods (e.g. API), and send data items [5]. Thus in the attack surface measurement, channels, methods and data items act as the system's resources and define the attack surface measurement.

2.4 REINFORCEMENT LEARNING

Reinforcement learning is learning what to do and how to map situations to actions to maximize the numerical reward. Reinforcement learning is defined not by characterizing learning methods, but by characterizing the learning problem [6].

2.4.1ELEMENTS OF REINFORCEMENT LEARNING

Policy, a reward function, a value function, and a model of the environment are the different elements of reinforcement learning.

Policy is the action to be taken based on the state of the system, a reward function is the value for the action taken, and a value function is the sum of all the rewards. The goal of reinforcement learning is to maximize the value function. The state of the system changes with respect to the environment. The action to be taken depends on the state and can be categorized under exploitation and exploration

Exploitation is the concept of choosing the action with a high reward value. Exploitation may have good results at present but cannot promise better results at the end of the play. Whereas exploration may not provide the system with better values at the current play but may lead to handsome results at the end of the play.

CHAPTER III

PROPOSED WORK

3.1 INTRODUCTION

The goal of this work is to design an architecture, that reduces the attack surface in self-service cloud computing. In addition to the components involved in SSC the new architecture will have a *virtual shield* (VS) that exists between the host virtual machine and meta domain as shown in fig 3.1.The SSC protocol will be modified to facilitate the interaction between MTSDs and the virtual shield.

3.2 SYSTEM BUILDING

Bootstrapping in the new architecture is similar to the SSC model [1]. The systems in the cloud are assumed to be equipped with TPM (Trusted Platform Module) and IOMMU (Input Output Memory Management Unit) hardware. During the system boot, the BIOS pass control to the boot loader. The boot loader loads the modified version of xen hypervisor. The hypervisor builds the Sdom0, domB and the virtual shield Vs0.

Once the Sdom0, DomB and the Vs0 are started, the Sdom0 waits for the client request. Once the client sends the request to Sdom0, the Sdom0 sends the request to DomB with the parameters send by the client. This ensures the security and privacy of the client. The DomB, then builds the client meta domain as in SSC.

The protocol used in the SSC architecture overcomes two issues as presented in the section 2.2, however the host virtual machine is not protected from malicious administrators and the guest virtual machines.

In this work, the virtual shield protect host the virtual machine.

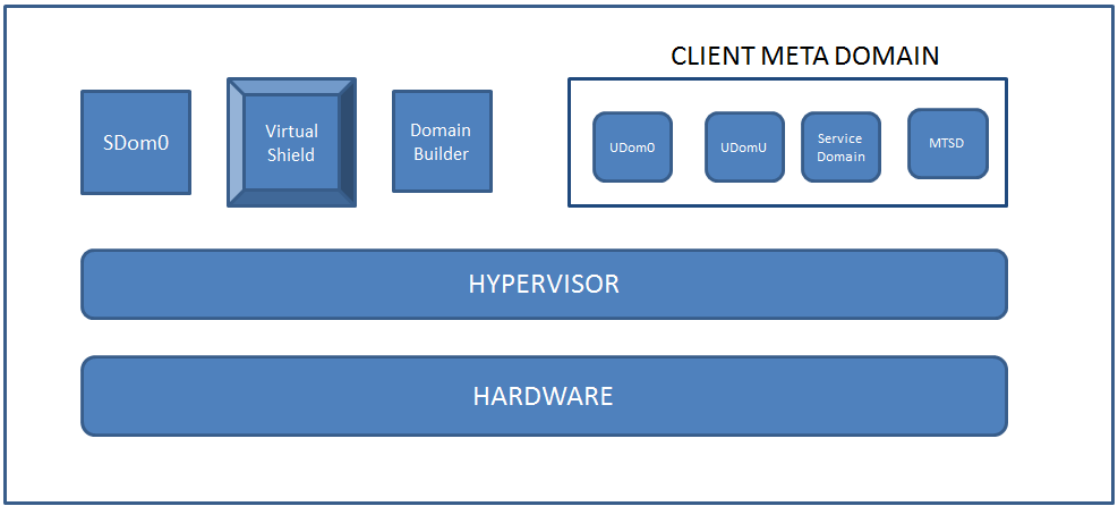

Figure 3.1 Self Service Cloud with Virtual Shield

3.3 COMPONENTS

1. SDom0

SDom0 is the system side administrative domain. This domain controls the client virtual machines. The start and stop of the client virtual machines is done by SDom0. Though this component has the same functionalities as SDom0 in SSC, it has additional capabilities which don't exist in SSC SDom0[1].

The SSC SDom0 has no privilege to view the state of the client virtual machines, i.e. the contents of virtual CPU, virtual memory etc. But in our proposed new architecture the SDom0 will be designed to have access if the client virtual machine is found to be malicious. The virtual shield provides these capabilities to SDom0 by providing the privileges to access the client virtual machines states.

2. Virtual Shield

The virtual shield is designed with different functionalities and security measurements. MSTD designed in the SSC are the key source of information to the virtual shield, which provides the information about the type of attack and the severity of the attack. In SSC the MSTD act as the regulatory compliance between client virtual machines and cloud providers. In SSC once the client virtual machines are identified to be misusing the cloud infrastructure for malicious activities the virtual machines are shut down and they lose its state.

In our architecture the virtual machines are not shut down immediately. Once the MSTD identify the client virtual machine to be malicious, it triggers the virtual shield with the information.

The information from the MSTD is used by the virtual shield for the reinforcement learning process designed to virtually configure the virtual machines to maximize security. The virtual shield holds a table with appropriate actions to be taken based on the state of the machine. Each

state has a reward value. The actions are the virtual configurations between the host virtual machine and the client virtual machine. Virtual shield holds one table for each virtual machine.

3. Domain Builder

DomB, the domain builder builds the client side Meta domain. Once the client sends the request to build the virtual machines, these parameters are send to the domain builder and virtual shield. Domain builder uses these parameters to build the client side Meta domain. The construction of Meta domain is similar to SSC, whereas the MTSDs are configured to trigger the virtual shield when the client misuses the cloud infrastructure.

4. Client Meta Domain

The Client Meta domain holds UDom0, UDomU, SDs and MSTD. All these components are assumed to have the same functionalities as in SSC, except the MTSD. The MTSD is modified to trigger the virtual shield when the guest virtual machines try to perform security attacks.

3.4 METHODOLOGY IN VIRTUAL SHIELD

One of the reinforcement learning methodologies will be used for the learning process in the virtual shield [6].

A Wide range of applications can be framed as reinforcement learning problems. The application in the virtual shield is framed to one of reinforcement learning tasks and provided with the method to learn.

The aim of the reinforcement learning problem is learning from interaction to achieve a goal. The decision maker is called the agent. The agent interacts with the environment, that is, everything that is outside the agent. This is a continuous process; the agent selects the actions and the

environment responds to those actions and provides new situations to the agent. The environment also provides the numerical rewards, which the agent tries to maximize over time. A task is defined as a configuration change, which is one instance of the reinforcement learning problem.

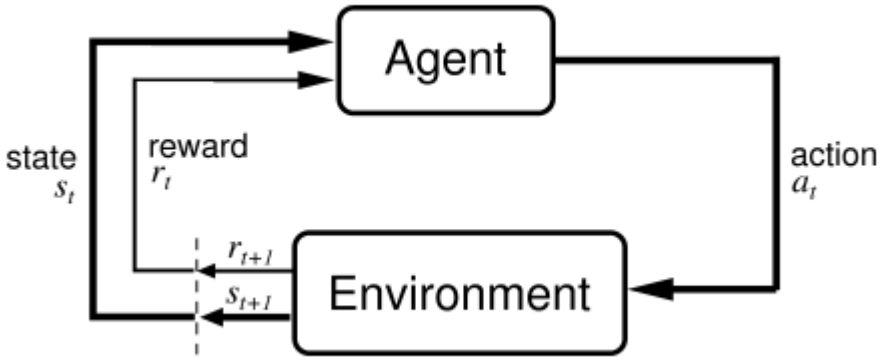

Figure 3.2 The agent-environment interaction in reinforcement learning

At discrete time steps, *t=0,1,2,3,……* the agent and the environment interact with each other. At each step *t*, the agent is provided with some representation of the environment's state $S_t \in S$, where *S* is the set of possible states and S_t is one of the states of **S**. $A(S_t)$ are the set of actions available in that state. After selecting the action $A_t \in A(S_t)$, the agent receives a numerical reward, $r_{t+1} \in R$ and enters a new state.

At each time step, a mapping from states to probabilities of selecting each passible action is implemented. This mapping is called the agent's policy $\prod_t$, where $\prod_t(S,A)$ is the probability that $A_t = A$ if $S_t = S$ where *S* is State and *A* is Action). In reinforcement learning the agent changes its policies as a result of experience. The agent's goal is to maximize the total amount of reward it receives over time. In our system the mutually trusted service domain is framed as the environment and the virtual shield is framed to be the agent.

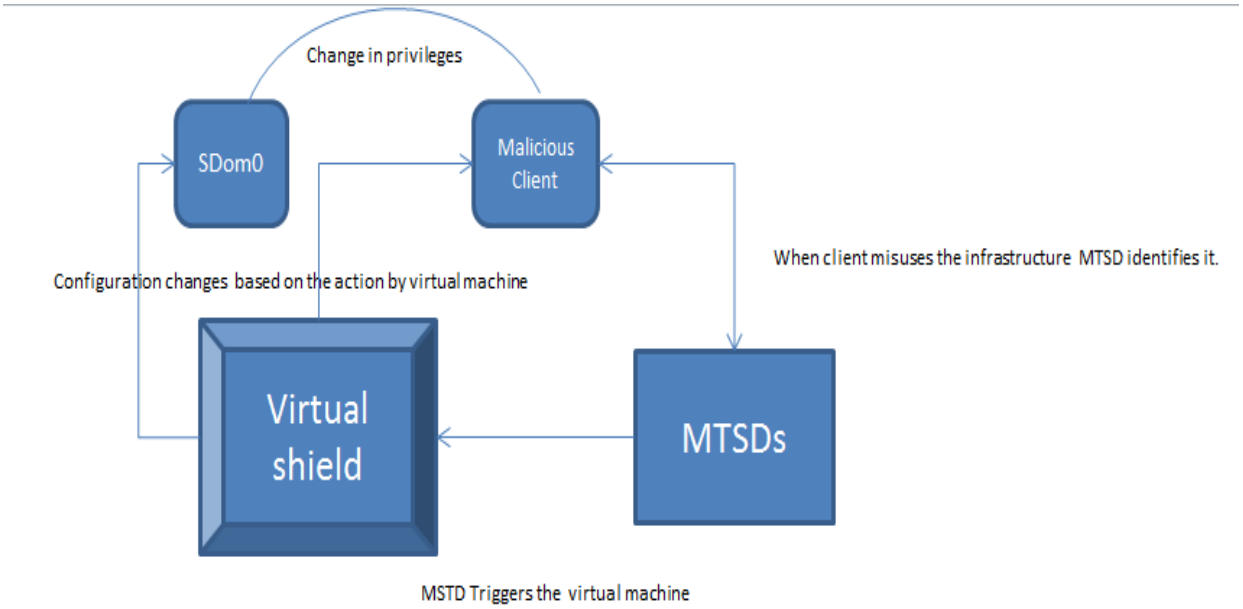

Figure 3.3 Proposed Architecture

CHAPTER IV

SIMULATIONS AND RESULTS

4.1 IMPLEMENTATION

CloudSim [14] is used to simulate the cloud environment. Cloudsim is a simulation environment to simulate the cloud architectures before actual deployment. Cloudsim provides java APIs to design the various elements of the cloud computing architecture. The underlying architecture contains different subsystems. Each subsystem is designed and simulated to satisfy the requirements of the whole architecture.

Different subsystems in the architecture includes

1. SSC SUB SYSTEM

2. MTSD SUB SYSTEM

3. VIRTUAL SHIELD SUB SYSTEM

4. ATTACK SYSTEM

5. CONFIGURATION SYSTEM

In the following section we discuss these subsections in detail.

4.1.1 SSC SUB SYSTEM

The SSC sub system is the main system which initializes the entire architecture. Upon request from the client, the administrative domain (Broker) in the SSC sub system requests the Data Center (Hypervisor) to allocate resources to the client.

During the initialization, the other subsystems are also activated or initialized.

The current client configuration will be written to the virtual shield.

The mutually trusted service domain is initialized with the different attack models and configuration parameters, which are in turn used to detect the malicious clients.

4.1.2 MUTUALLY TRUSTED SERVICE DOMAIN (MTSD)

During the client initialization, the mutually trusted service domain is also initialized with the different attack models to check the client's attacks.

The Mutually Trusted Service Domain periodically checks the network packets transmitted to identify the malicious clients. During this process if the MTSD identifies that the client is misusing the application, it notifies the virtual shield with the attack type. This attack type is used to calculate the score of the current configuration.

4.1.3 VIRTUAL SHIELD

The mutually trusted service domain triggers the virtual shield periodically with the activities of the client. If the MTSD identifies that the client is misusing the cloud infrastructure, depending upon the type of attack, severity of the attack and the existing configuration, it triggers the virtual

shield and the virtual shield calculates the score of the existing configuration and requests the administrator to change the configuration of the client.

The Virtual shield uses the simple reinforcement learning mechanism to allocate the different configuration parameters to the client.

The reinforcement learning mechanism performs exploration and exploitation to allocate the configurations. During exploration the virtual shield selects the random configurations from the configuration database. Once all the allocated configurations for the client have been explored, it uses the exploitation mechanism to fetch the configuration which has the highest score.

Each individual configuration is assigned a default score initially. The type of attack determines the calculated score for the configurations.

4.1.4 CONFIGURATION SYSTEM

Configuration defines the properties of the virtual machines such as computing capacity in terms of million instructions per second, image size, memory size, number of cpus, and bandwidth. The configuration system is a database which holds the different configuration parameters for different clients. It has the different configurations for different type of attacks. The virtual shield allocates these configurations to the clients by analyzing the existing configuration and type of attack the client performed.

Different parameters in the configurations include: MIPS, IMAGE SIZE, MEMORY SIZE, CPUS, and BAND WIDTH.

MIPS (Million instructions per second) define the number of instructions to be executed per second.

Image size defines the size of the operating system image.

Memory size defines the size of the internal memory.

CPUs define the number of cpus required by the virtual machine.

Bandwidth defines the network bandwidth (number of bits transmitted per second).

4.1.5 ATTACK SYSTEM

The attack system is a database which holds different attacks metrics. These attack metrics are used by the Mutually Trusted Service Domain to identify the malicious client and notify the virtual shield with the type of attack and severity of the attack. There can be different metrics to identify the attack types. In our architecture for the purpose of simulation we used a 8 bit binary coded value to identify the attack type and severity of the attack.

10000001 – First four bits of the code determines the type of attack and last four bits are used to determine the severity of the attack.

Here 1000 determines the type of attack in the database and 0001 determines the severity of the attack.

These metrics are used by the virtual shield along with the existing configuration score to determine the new score of the current configuration.

4.2 COMMUNICATION PROTOCOL

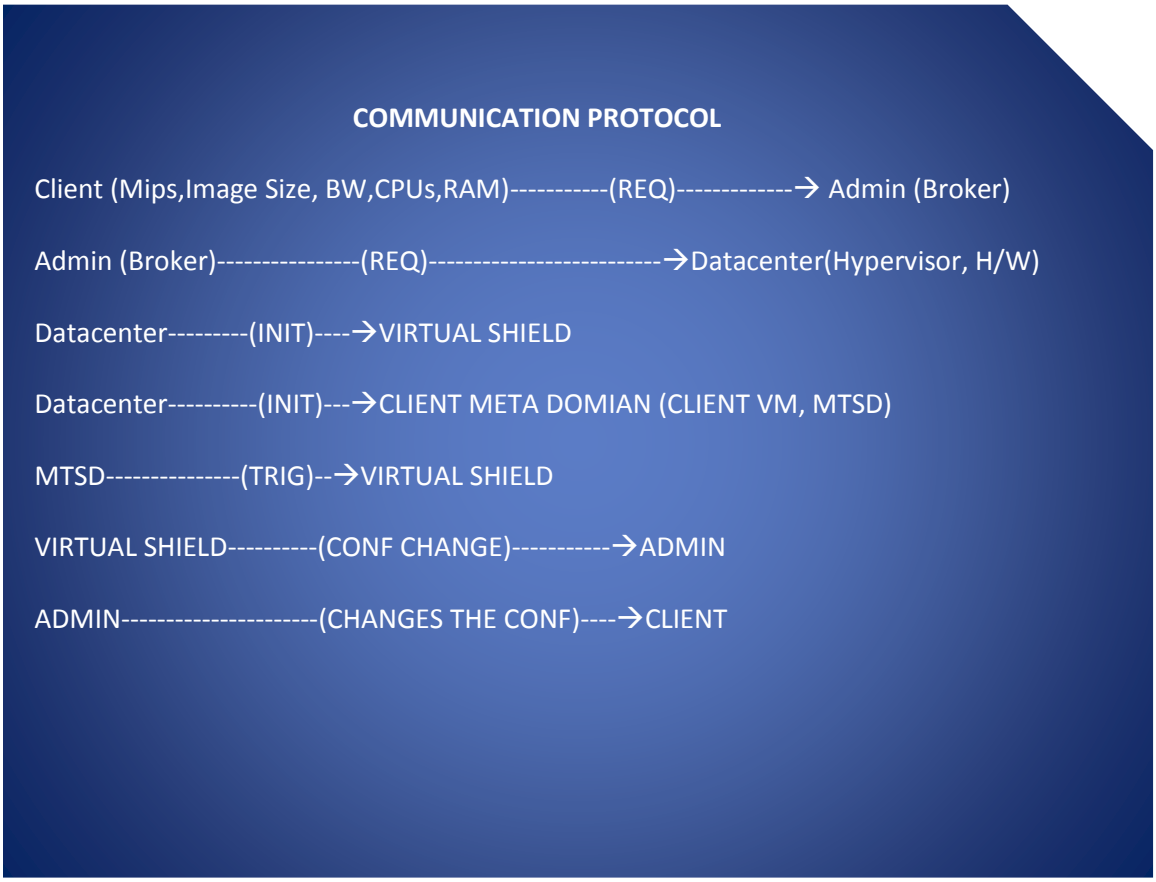

Figure 4.1 Communication Protocol

The communication protocol in the above figure4.1 explains how each subsystem in the architecture interacts with each other. After the initialization of Sdom0, the client requests the system side administrative domain for the virtual machine by passing the configuration parameters.

The Dom0 requests the datacenter i.e. the Hypervisor to provide the requested resources to the client.In this process, the hypervisor initializes the virtual shield and clients meta domain.

If the client tries to misuse the resources, MTSD triggers the virtual shield. The Virtual shield in turn calculates the scores for the current configuration based on the type of the attack and severity of the attack. It also changes the current configuration to another configuration which has highest score during exploitation.

This architecture on which this protocol executes is shown in figure 3.3

4.3 SCORE CALCULATION:

The last four bits of the binary code in the attack system is used to calculate the score for the current configuration. The first four bits determines the type of attack. The value of the type of attack is multiplied by the value of the last four bits to generate the score to be subtracted from the current score. The figure 4.2 explains the score calculation for the current configuration.

10000001
1000 ----------> = 8
0001 ---------> = 2

S(c) ------> Conf Score
S(a) ----> Attack Score
T(a) ----> Type
Se(a) ----> Severity
S(N) -----> New Score

S(a) = T(a) * Se(a)
S(a) = (8*2) =16
S(N) = S(c) - S(a)
S(c) = S(N)

Figure 4.2 Score Calculation

4.3.1 VIRTUAL MACHINE TERMINATION

The mutually trusted service domain periodically checks the clients meta domain for attacks. These periodical updates are notified to the virtual shield to calculate the scores for individual configurations. The individual scores of the allocated configurations to the client are aggregated to identify the overall score of the client's virtual machine. This aggregated score is used to determine the threshold for the client termination. Once the virtual shield identifies the total score is less the threshold designed by the cloud provider or the administrator, the client's virtual machine is terminated.

The client is notified every time the configuration changes. If the client still tries to misuse the cloud infrastructure, the over al score eventually decreases and finally results in the termination of the clients virtual machine. The threshold is defined by the cloud provider for each and every client virtual machine. If the overall score of the client virtual machine is less than the threshold, the virtual shield informs the administrator to terminate the client's virtual machine. This process is explained in figure 4.3.

$$O(c) = \sum_{i=0}^{n} Si(c)$$

If O(c) < T(c)
Terminate

O(c) ---->Client Score
T(c) ----->Threshold
Si(c) ----->Scores of individual configurations
n -------->number of configurations

Figure 4.3 Terminating Condition

TYPE OF ATTACK	SEVERITY OF ATTACK	ATTACK TYPE CODE	ATTACK SEVERITY CODE
Denial Of Service	VERY HIGH	10001111	1111
VM Escape	HIGH	01000011	0011
Inter-VM Attacks	HIGH	00100011	0011
Communication Attacks	MODERATE	00010001	0001

Table 4.1 Attacks Information

We assumed six different types of attacks by the client's virtual machine.

Denial of Service:

Denial of service or Distributed denial of service attack is an attempt to make a machine or network unavailable to the intended users. In our architecture this type of attack can be performed my clients virtual machine by sending multiple requests to the hypervisor and thereby consuming more resources [10].

VM Escape:

Security between the Host and VM is another area which introduces a threat called VM escape [9]. There are different zones in the cloud and even if a single VM is attacked it poses a threat to the whole system. If a malicious user can get access to a VM then he can directly communicate with the Hypervisor and can get complete access to all the VM's.

Inter VM attacks:

Inter VM attacks refer to attacks launched from one virtual machine to another co-residing virtual machine directly, typically by bypassing the virtual machine monitor (VMM) [13].

Communication attacks:

These are attacks launched by the virtual machines on the network channel. Routers and switches may be compromised due to communication attacks

REINFORCEMENT LEARNING ALGORITHM FOR VIRTUAL SHIELD

EXPLORATION

```
Virtual Shield (Exp)
If Exp==0
   Generate Random Number
   Choose Configuration State
   Set Current State = Configuration State (Random Number)
   Configuration State. Remove (Random Number)
    If (Configuration State is Empty ())
      Exp=1;
```

EXPLOITATION

```
Else
      Get Score (State Zero)
      Set B to 0
      For All States in Virtual Shield
            Set Y to X
             Set X to MAXIMUM of X, Get Score (B)
             If (X! = Y)
            Set new State to Virtual Shield. State (B).
            Increment B
```

SCORE GENERATION

```
Generate Scores ()
If Attack= True
Current State. Score= Current State. Score -ErrorCodeScore;
Change Configurations
Else
State. Get (index).score=state. Get (index).score+3
```

4.4 RESULTS

CloudSim [14] was used to simulate the entire environment. Cloudsim provides java APIs to design the various elements of the cloud computing architecture. We defined three different classes for this simulation.

1. CLIENT CLASS
2. MSTD CLASS
3. VIRTUAL SHIELD CLASS

These classes are explained as the subsystems in section 4.1

The following graph determines how secure the client virtual machine is. Once the client virtual machines score reaches below the threshold, the virtual machine is terminated.

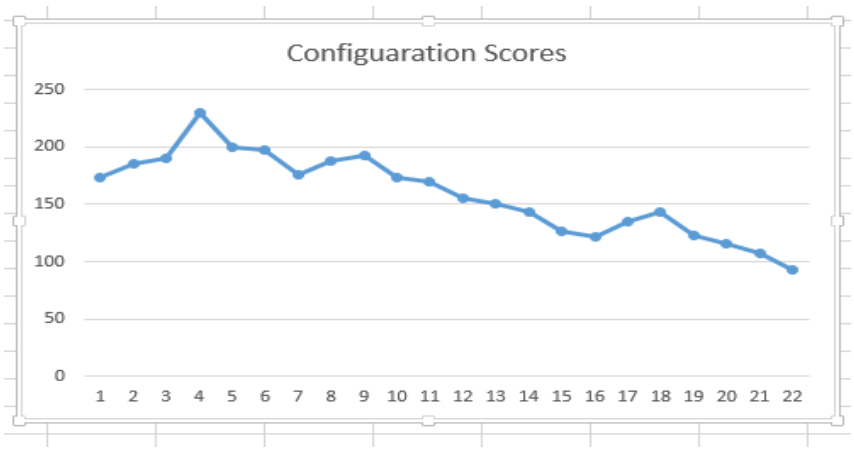

Figure 4.4 Configuration Graph

TIME IN MINUTES	CONFIGURATION SCORES
1	173
2	186
3	190
4	230
5	200
6	198
7	176
8	188
9	193
10	173
11	170
12	156
13	150
14	143
15	127
16	122
17	135
18	143
19	123
20	116
21	107

Table 4.2 Configuration Scores

Table 4.2 explains the change in configuration for every minute. This can be observed from the table that if the client virtual machine tries to attack the total configuration score of that client virtual machine decreases. On the other hand the score increases if the client virtual machine acts safely.

The above graph shows the activity of the client. It can be observed that at the 22nd minute the total configuration score is less than the threshold which results in termination of the virtual machine.

As we can observe from the table 4.2, the score increases till the 3rd minute and decreases at 4th minute. This indicates that the cloud infrastructure is misused or the client is malicious. The gradual decrease in the score can be observed from 9th to 16th minute, this indicates that the client is malicious. Similarly from the 16th minute till 18th minute the score increase, this indicates that the client is not misusing the cloud infrastructure.

CHAPTER V

CONCLUSIONS

Self-service cloud computing reduces the attack surface of the traditional cloud architecture by transferring most of the privileges to the clients Meta domain. However this architecture is not designed to protect the inter virtual machine attacks and clients vm attacks on the administrative domain and hypervisor.

In the proposed architecture (SSC with Virtual Shield), the client side attacks have been mitigated by dynamically configuring the virtual machines based on the type and severity of the attacks performed by the clients.

SSC with virtual shield is a new computing model designed to protect the host virtual machine from various attacks by the guest virtual machines. The proposed new design has the capability to shift the privileges between the system side administrative domain and client side administrative domain. This dynamic configuration of virtual machines reduces the attack surface and makes the cloud more secure.

A simple reinforcement learning algorithm has been used to calculate the scores and configuration changes. This algorithm uses exploitation and exploration to perform this operation.

If the client tries to misuse the cloud infrastructure, the configuration changes according to the information present in the configuration subsystem. The simulation and results section explains the configuration changes and virtual shield termination.

In the proposed architecture, the reinforcement learning algorithm has been used to make the virtual shield learn from the environment. This algorithm holds good for a minimum number of client virtual machines, since a single virtual shield runs this algorithm to calculate the scores. The overhead on the virtual shield increases to handle multiple clients and multiple virtual machines. Moreover if the virtual shield fails, there is no way to protect the entire system from the attacks of the client. The virtual shield can fail because of hardware problem or may be due to the extra over head in handling multiple virtual machines.

The proposed architecture can be enhanced by removing the single point of failure by having a backup virtual shield called a Stand-by Virtual Shield which performs backup tasks by snapshotting. Snapshotting is the process of identifying the virtual machines state and securely storing the states in external devices. If the active Virtual Shield fails, the stand-by virtual shield can be made active. Multiple clients can be simultaneously handled by introducing the concept of multiple VM clusters in the virtual shield. Multiple virtual machines are associated with single virtual shield. This works by replacing the reinforcement learning algorithm with map and reduce functions running on the cluster of VM's. This removes the overhead on the system and can provide more accurate results by analyzing the system log files for different kind of attacks.

REFERENCES

[1] Shakeel Butt, H.Andres Lager-Cavilla, Abinav Srivastava and Vinod Ganapathy, "Self Service Cloud Computing", *ACM Conference on Computer and Communications Security*, pages 253-264, October 2012.

[2] Nils Gruschka and Mieko Jensen, "Attack Surfaces: A taxonomy for Attacks on Services" *3rd International Conference on Cloud Computing*, pages 276-279, 2010.

[3] Asoke K Talukder, "Analyzing and Reducing the Attack Surface for a Cloud-ready Application" *Indo-US Confeence on Cybersecurity, Cybercrime and Cyberforensics*, August 2009.

[4] Trend Micro White Paper, "Cloud Computing Security", website: http://www.securecloud.com/cloud-content/us/pdfs/business/whitepapers/wp_cloudsecurity-unlock-opportunities.pdf , May 2010

[5] Pratyusa K. Manadhata and Jeannette M. Wing, "A Formal Model for a System's AttackSurface" *Technical Reports HPL-2011-115*, website: http://www.hpl.hp.com/techreports/2011/HPL-2011-115.html, 2011.

[6] Richard S. Sutton and Andrew G. Barto, " *Reinforcement Learning An Introduction*" A Bradford Book, 1988

[7] Shyam Patidar, Dheeraj Rane and Pritesh Jain , "A Survey Paper on Cloud Computing" *Second International Conference on Advanced Computing & Communication Technologies*, pages 394-398, 2012

[8] Flavio Lombardi and Roberto DI pietro ," Secure virtualization for Cloud computing" *Journal of Network and Computer Applications*, volume 34, issue 4, pages 1113-1122, June 2011

[9] Farhan Bashir Shaikh and Sajjad Haider, " Security Threats in Cloud Computing" *6TH International Conference on Internet Technology and Secured Transactions*, pages 214-219, December 2011

[10] S VivinSandar and SudhirShenai , " Ecnomic Denial of Sustainbility in Cloud Services using HHT and XML based DDoS Attacks*" International Journal of Computer Applications*, volume 41, issue no-20, pages 11-16, March 2012.

[11] Guojun Wang, Qin Liu and Jie Wu, " Hierarchical Attribute-Based Encryption for Fine- Grained Access Control in Cloud Storage Services" *17th ACM conference on Computer and communication security*, pages:735-737, 2010.

[12] Balachandra Reddy Kandukuri, Ramakrishna Paturi V and Dr. Atanu Rakshit, "Cloud Security Issues" *IEEE International Conference on Services Computing*, pages: 517-520, 2009.

[13] Tien-Hao Tsai, Yen-Chung Chen, Hsiiu-Chuan Huang, Pei-Ming Huang and Kuo-Sen Chou, "A Practical Chinese Wall Security Model in Cloud Computing" *Network Operations and Management Symposium*, pages:1-4, 2011

[14] Rodrigo N. Calheiros, Rajiv Ranjan, Anton Beloglazov, Cesar A. F. De Rose, and Rajkumar Buyya, *CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms*, *Software: Practice and Experience (SPE)*, Volume 41, Number 1, Pages: 23-50, ISSN: 0038-0644, Wiley Press, New York, USA, January, 2011.

VITA

Balaji Ganesula

Candidate for the Degree of

Master of Science

Thesis:

Major Field: Computer Science

Biographical:

Education:

Completed the requirements for the Master of Science in Computer Science at Oklahoma State University, Stillwater, Oklahoma in Dec, 2013.

Completed the requirements for the Bachelor of Science in Computer Science at SRM University, Chennai, India in May, 2010.

Experience:

Graduate Teaching Assistant, Department of Zoology Aug 2012 – Dec 2013
Oklahoma State University : Stillwater,OK
Working as a web application developer in Zoology Dept.
Supported and developed BIOL1114 website.
Administered and maintained the database for BIOL1114.
Graduate Teaching Assistant, Department of Computer Science Jan 2012 – May 2012
Oklahoma State University : Stillwater,OK
Graded assignments and maintained confidential student information.
Taught undergraduate courses in computer networking.
Asst. Systems Engineer, Tata Consultancy Services Nov 2010 – Aug 2011
Chennai, India
Completed training in java web application development using JSPs and Servlets.
Played an active role in creating a web application for an insurance company.
Worked in a team to develop and support an application for a mobile manufacturing company, which is based on Siebel Technology.