Cooperative Trust Framework for Cloud Computing Based on Mobile Agents

Hicham Toumi[1], Amal Talea[2], Bouchra Marzak[3], Ahmed Eddaoui[4], Mohamed Talea[5]

[1, 3, 5] Information Processing Laboratory, Department of Physical, University Hassan II Casablanca, Morocco
[2,4]Department of Mathematics and Computer Science, University Hassan II, Casablanca, Morocco
toumi.doc@gmail.com, amal.talea@gmail.com, marzak8bouchra@gmail.com, ahmed_eddaoui@yahoo.fr, taleamohamed@yahoo.fr

Abstract: Cloud computing opens doors to the multiple, unlimited venues from elastic computing to on demand provisioning to dynamic storage, reduce the potential costs through optimized and efficient computing. To provide secure and reliable services in cloud computing environment is an important issue. One of the security issues is how to reduce the impact of any type of intrusion in this environment. To counter these kinds of attacks, a framework of cooperative Hybrid intrusion detection system (Hy-IDS) and Mobile Agents is proposed. This framework allows protection against the intrusion attacks. Our Hybrid IDS is based on two types of IDS, the first is for the detection of attacks at the level of virtual machines (VMs). The second is for the network attack detection and Mobile Agents. Then, this framework unfolds in three phases: firstly, detection intrusion in a virtual environment using mobile agents for collected malicious data. Secondly, generating new signatures from malicious data which were collected in the first phase. The thirdly, dynamic deployment of updates between clusters in a cloud computing, using the newest signatures previously created. By this type of close-loop control, the collaborative network security management system can identify and address new distributed attacks more quickly and effectively.

In this paper, we develop a collaborative approach based on Hy-IDS and Mobile Agents in Cloud Environment, to define a dynamic context which enables the detection of new attacks, with much detail as possible.

Keywords: Cloud Computing, Hy-IDS, Mobile Agents, Collaborative, Signatures**.**

1. Introduction

Cloud Computing is the latest developments of computing models after distributed computing, parallel processing and grid computing. It is an innovative computing model in which resources are provided as a service over the Internet, alleviating users from the responsibility of buying and managing computing infrastructure. It also provides a shared pool of resources, including data storage space, networks and computer processing power. These components can be rapidly deployed, provisioned, implemented and scaled up or down. It provides a model of allocation and consumption on demand. Cloud allows improved to flexibility, scaling and availability, and provides the potential for cost reduction through optimized and efficient computing. At the same time, the transformational nature of the cloud is associated with significant security and privacy risks. Although the cloud model is designed to reap numberless benefits for all cloud stakeholders including cloud providers (CPs), cloud consumers (CCs), and service providers (SPs), the model still has a number of open issues that impact its credibility [1].

But, the intrusion detection or confidentiality of data over Cloud is one of the glaring security concerns. The fast growth of cloud computing technology introduces more of the vulnerabilities. Security is considered to be one of the most critical aspects in cloud computing environment due to the confidential and important information stored in the cloud [2].

Network security appliances, such as Intrusion Detection Systems (IDS) is widely deployed in advantage points and play an important role in protecting the network from attacks. That’s why, it is nowadays widely deployed for securing critical IT-Infrastructures. It is based on the protected environment, IDS can be classified into host-based intrusion detection systems (HIDS), network-based intrusion detection systems (NIDS), or distributed intrusion detection systems (DIDS) which contain both types of sensors. Due to different deployment mechanisms, we can distinguish different types of IDS; IDS can be categorized as software-based IDS, hardware-based IDS, and VM-based IDS [3]. Most of these appliances work without collaboration, their detection results are isolated and cannot be collected and analyzed systematically. Therefore, we thought of a new security policy that allows the detection of distributed attacks such as deny of service (DoS) and Distributed Denial of Service (DDoS) [2].

In this paper, we will deepen the development of our approach based in principle on the cooperation of the Hybrid Intrusion Detection System (Hy-IDS) and mobile agents. The cooperation between our Hy-IDS and mobile agents present what is called a framework. This framework allows to reach three objectives: the first, detection intrusion in a virtual environment using mobile agents for collecting malicious data. The second, generating new signatures from malicious data, which were collected in the first phase. The third, dynamic deployment of updates between clusters in a cloud computing, using the newest signatures previously created.

The rest of this paper is organized as follows: The section II presents theoretical background and discusses some related works in the area of Mobile Agent-based IDS and NIDS. The section III forms the core of this paper explains and describes in detail our approach. Whereas the proposed framework is discussed in section IV. Finally we give conclusion, perspective and references in section V.

2. Theoretical Background and Related Work

In this section, we start with theoretical background include cloud computing and Virtualization technologies as the first part, and Related Work as a second part.

2.1 Cloud computing

As the Cloud combines many well-known technologies and leads to complicated IT systems and networks. For this reason, we can consider cloud computing as a model that allows practically, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services), that can be rapidly provisioned and released with minimal management effort or service provider interaction.

Cloud computing uses virtualization, service-oriented software, and grid-computing technologies among others. It allows accessing resources and services offered by servers from different places. Therefore, it is a model of distributed computing [4]. It is also a new large-scale distributed computing model. Virtualization, instantaneous deployment, broadband networks and other key technologies are applied in the cloud computing.

The Cloud Computing is undergoing an incontestable success, which could be indeed compromised by concerns about the risks related to potential misuse of this model aimed at conducting illegal activities. To provide secure and reliable services in cloud computing environment is an important issue. Then, There have been a great deal of inherent issues in cloud computing such as the security of data, the management of vulnerability, the system of disaster recovery, the process of business continuity and identity management [5]. The prime challenges of the process are as follows [6]: 1) Confidentiality, 2) Auditability, 3) Control Over Data Lifecycle, 4) Privileged User Access, 5) Lack of Standards and Interoperability and 6) Multi Tenancy. Then, there are numerous security issues in cloud computing as it encompasses many technologies including networks, virtualization, load balancing, operating systems, transaction management, resource scheduling, concurrency control and memory management[7]. Therefore, security issues for many of these systems and technologies remains very much current issues in a cloud computing environment.

One of the initial steps toward cloud computing is incorporating virtualization, which is separating the hardware from the software. Then, the underpinning for the majority of high-performing clouds is a virtualized infrastructure. Virtualization is used more broadly to pool infrastructure resources, virtualization can also provide the basic building blocks for your cloud environment to enhance agility and flexibility [8].

2.2 Virtualizing the Cloud Computing Infrastructure

Virtualization is a term that refers to the abstraction of computer resources. It enables us to use one physical server with the capability of delivering the performance of multiple servers [9]. This technology enables customers to run multiple operating systems concurrently on a single physical server, where each of the operating systems runs as a self-contained computer [10]. More recently, virtualization at all levels became important again as a way to improve system security, reliability and availability, reduce costs, and provide greater flexibility. There are three types of virtualization: Para virtualization, Container virtualization and the last Full virtualization.

A. **Para virtualization:** It is a technique in which the guest operating system is aware that they are operating directly on the hypervisor instead of the underlying hardware [11]. In Para virtualization a Para virtualization supporting hypervisor is installed on the host operating system which runs over the underlying hardware [12].

B. **Container virtualization:** Container virtualization is a technique in which each operating system kernel is modified to load multiple guest operating systems [13]. The kernel provides proper management of the underlying resources to isolate one container activity from the other.

C. **Full virtualization**: In full virtualization hypervisor supporting the full virtualization technique is installed directly over the underlying hardware. In this approach, the VMM is also called virtual machine manager and runs on top of a host operating system [14].

The adoption of virtualization in cloud computing brought up several advantages compared to traditional computing infrastructure, in which one resource was allocated to a single VM at a time [15]. Since a virtual environment (VE) composed of numerous clients VMs that are controlled by VMM to ensure fair scheduling, memory and resource allocation etc. The VMM itself is running on the physical infrastructure by utilizing the host OS [16]. The VE is similar to the OS environment, where multiple processes are running on a single OS which is responsible for monitoring and managing the processes [17].

2.3 Mobile agent technology in cloud computing

The Mobile Agent has its applications in many areas including network management, mobile computing, information monitoring, searching for information, remote software management and others. Mobile Agents enhance the performance in these areas by providing the following services [18]:

- Efficiency and reduction of network traffic: The mobile agent can operate asynchronously and independent of the sending program. Mobile agents consume fewer network resources since they move the computation to the data rather than the data to the computation. They allow the processing to be performed locally, instead of transmitting the data over a network.
- Interaction with real-time entities: Real-time entities require immediate responses to changes in their environment. Controlling these entities from across a potentially large network will incur significant latencies. Mobile agents offer an alternative to save network latency.
- Convenient development paradigm: The design and construction of distributed systems can be made easier by the use of mobile agents. There are three basic needs for Mobile Agents to achieve these goals, the Mobile Agent Program, Mobile Agent Platforms and Mobile Agent Creator.
- Life cycle of mobile agent: Creation, Cloning, Dispatching or Migration, Retraction, Activation, Deactivation and finally Disposal have been carried out through java.

2.4 Intrusion detection systems (IDS)

Intrusion Detection Systems (IDS) have become an active area of research to develop reliable and effective solutions to detect and counter the increasing number of intrusions in systems’ and networks’ infrastructures cloud computing.

An IDS can be either host-based, if it monitors system calls or logs, or network-based if it monitors the flow of network packets. However, the new generations of IDSs are usually a combination of these two approaches. Another important distinction is between systems that identify patterns of traffic data presumed to be malicious, and systems that compare activities against a ‘normal’ baseline (anomaly detection systems) [19]. Then, when an intrusion is discovered by IDS, typical actions to perform would be logging relevant information to a file or database, generating an email alert, or generating a message to an administrator. However, some forms of automatic reaction can be implemented through the interaction of IDSs and access control systems such as firewalls.

The Intrusion Detection Service (IDS) increases a Cloud’s security level by providing two methods of intrusion detection. The first method is a behavior-based method which dictates how to compare recent user actions to the usual behavior. The second approach is a knowledge-based method that detects known trails left by attacks or certain sequences of actions from a user who might represent an attack. The audited data is sent to the IDS service core, which analyzes the behavior using artificial intelligence to detect deviations. This has two subsystems namely analyzer system and alert system. In order to detect the intruders the following techniques should be implemented in either HIDS or NIDS [20] [21] [22].

In [23] for improving IDS performance the authors proposed an efficient model that used multithreading technique in Cloud environment to handle large number of data packet flows. The proposed multi-threaded NIDS is based on three modules named: capture module, analysis module and reporting module. The first one is charged of capturing data packets and sending them to analysis part which analyzes them efficiently through matching against pre-defined set of rules and distinguishes the bad packets to generate alerts. Finally, the reporting module can interpret alerts and immediately make alert report.

2.5 Signature generation algorithm

To prevent systems from new attacks, the IDS should be quickly updated. However, attacker instead of finding new types of attack tries to remain unnoticed in the evading system by using signature. If we take one of the types of IDS as NIDS; for real time evasion IDS (e.g., NIDS) is created using the signature generation algorithm (e.g., Apriori Algorithm, Signature Apriori Algorithm). The aim of evasion is not to break the NIDS system but to make system sturdier. Different sessions of attacks are given as input to the signature generation algorithm. According to support and confidence value rule is generated by the signature generation algorithm. These rules are given to NIDS. When an attack is generated for which signature is stored in database NIDS, it generates an alarm. If NIDS failed to generate alarm means evasion is successful. Therefore, we found out different types of evasion [2][24].

2.6 Relevant Works and Limitations

In the literature there are few works that use IDS, NIDS (Snort and signature apriori algorithm) and mobile agents in the cloud computing. In this section, we present four works, the first work is based on Snort combined with a signature apriori algorithm. The second work is based on IDS and mobile agents. The third proposes a trust model for cloud architecture which uses mobile agent. Finally, another work based on mobile agents.

The first work presented by Chirag N. Modi et al, combine Snort (Snort-Home page N.d.) and signature apriori algorithm in their NIDS module. The objective of this approach is to reduce impact of network attacks (known attacks as well as derivative of known attacks). The network may be external network or internal network. Snort will monitor those network packets and allow/deny them based on the configured rules. Also, captured packets, partially known attack signatures (stored in known signature database) and support threshold are given as input to the signature apriori algorithm. Using given input, signature apriori generates new possible signatures and updates them as rules in Snort. Therefore, derivative attacks can be detected by Snort [25]. However, this work is unable to detect intrusion at the hosts, and Distributed denial of service attacks (DDoS).

The second work, A.V. Dastjerdi et al. they tried to offer a line of defense by applying Mobile Agents technology to provide intrusion detection for Cloud applications regardless of their locations. These researchers build up a robust distributed hybrid model scaled, flexible and cost effective method based on mobile agents (MA). VMs are attached to MA which collects evidences of an attack from all the attacked VMs for further analysis and auditing. Then, they have to correlate and aggregate that data to detect distributed attacks [26]. This kind of work is limited to the detection of attacks at machines. They did not think to monitor network traffic simultaneously.

The third work, In [27] the authors propose a trust model for cloud architecture which uses mobile agent as security agents to acquire useful information from the virtual machine which the user and service provider can utilize to keep track of privacy of their data and virtual machines. In this work, Security agents can dynamically move in the network, replicate itself according to requirement and perform the assigned tasks like accounting and monitoring of virtual machines for monitoring virtual machine integrity and authenticity.

The fourth work, facing new application scenarios in Cloud Computing, the IDS approaches product several problems. Then, IDS management is an important capability for distributed IDS solutions, which makes it possible to integrate and handle different types of sensors or collect and synthesize alerts generated from multiple hosts located in the distributed environment. S. Roschke, et al. They summarize several requirements for deploying IDS in the Cloud and propose an extensible IDS architecture for being easily used in a distributed cloud infrastructure [28].

After a thorough study of various security policies, we found the need for collaboration of several security policies. This collaboration is mainly based on mobile agents. Then we exploit mobile agents for security against intrusion attacks and at the same time as a communication tool between different layer of cloud computing. That’s why, we combine between the strengths of these previous works in our approach. We will argue in the next section that this collaboration has several advantages.

3. Collaborative Trust Framework Based on Cloud Computing: Hy-IDS and Mobiles Agents.

In the previous work [2], we presented introduction to a new architecture of cloud computing based on IDS, Signature Generation Algorithm and mobile agents. Then, in this section are first presented the objectives of the proposed framework, its overall architecture, highlighting its four main layers and overall functioning. Finally, we explain the role of each component and how it will react in case of attack.

3.1 Objectives of the framework

The objectives of our framework are grouped into three main Points as follows:

- Intrusions detection in a virtual environment using mobile agents in order to collect malicious data.
- Generating new signatures from malicious data, which were collected in the first part.
- Dynamic deployment of updates between clusters in a cloud computing, using the newest signatures previously created.

3.2 Our proposed hybrid framework

3.2.1 Proposed model of cloud computing

As shown in Figure.1, we define cloud architecture with a front-end and back-end. Front end is connected to both external network as well as internal network. It is presented in the figure 1 by the Cloud layer.

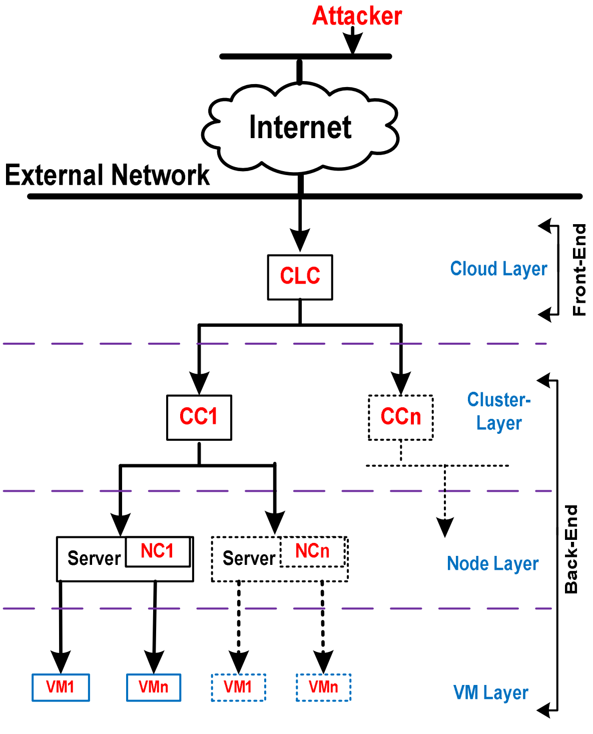

Figure 1. Proposed model of cloud

Then, Cloud users are able to communicate with Cloud via front end. Back-end consists of computer hardware and software that are designed for the delivery of services. It allows treatment of the user's query and executes it for allowing to access VM instances. Then, it is presented in the figure 1 by the Cluster-Layer, Node-Layer and VM-Layer.

The Cloud Controller (CLC) provides EC2-compatible interfaces, as well as a web interface to the outside world. The CLC acts as the administrative interface for cloud management and performs high-level resource scheduling. Only one CLC can exist per cloud and it handles authentication, accounting, reporting.

The Cluster Controller (CC) acts as the front end for a cluster within a cloud computing and communicates with the Cloud Controller and Node Controller. It manages instance (i.e., virtual machines) execution and Service Level Agreements (SLAs) per cluster.

The Node Controller (NC) at level of physical server; it hosts the virtual machine instances and manages the virtual network endpoints. Whereas, there is no theoretical limit to the number of Node Controllers per cluster.

3.2.2 Building a solution framework

Now, after the presentation of the cloud architecture, we proceed to the establishment or distribution of the components of our framework on this architecture according to our strategy protection. Then, the general architecture of our framework, shown in Figure 2, is divided into four main layers interact.

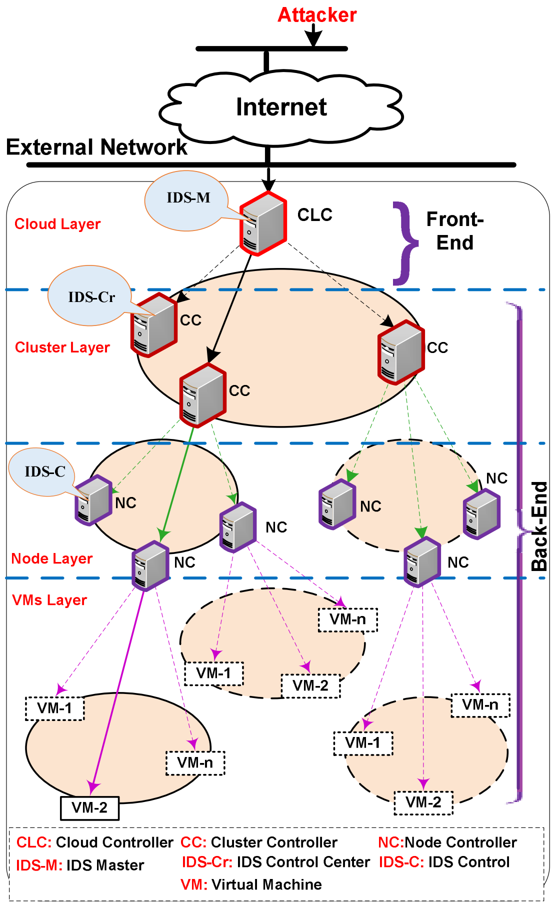

Figure 2. The Hierarchy of our cloud computing

Our Hybrid Intrusion Detection System (Hy-IDS) combines Intrusion Detection System Control (IDS-C) and Intrusion Detection System Center (IDS-Cr), which are placed in the back-end. Finally, Intrusion Detection System Master (IDS-M), which is placed in the front-end.

IDS-C: VMs are further managed by hypervisors, also known as Virtual Machine Monitor (VMM) and are basically installed on server hardware. Thus, as shown in figure 3, we use VMM in our framework to ensure a new level of trust in the VMs. Then, we place the components of IDS-C at the level of nodes (physical server) for monitoring virtual machines. For more details, we place IDS-C at the level of VMM. At the same time, we place specific static agent detectors (SA) at the level of VMs. Our IDS-C is based on the cooperation of IDS with the living environment of mobile agents named Agents Agency (AA).

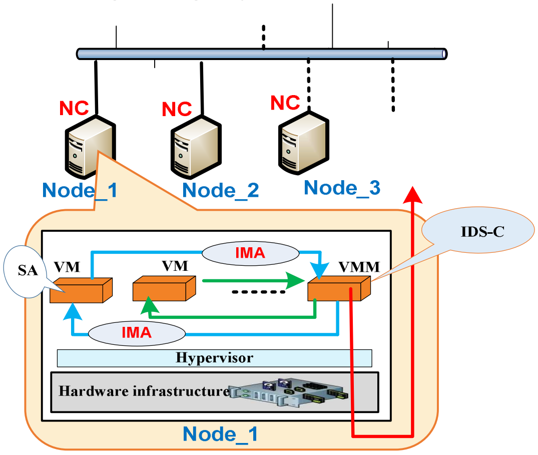

Figure 3. SA and IDS-C over the nodes

IDS-Cr: it installed in the front-end Cluster for the monitoring of nodes. It also generates new signatures. It consists of an Intrusion Detection System (IDS) and Signature Generation Algorithm (SGA).

IDS-M: it is placed in the front-end Cloud for the monitoring of Clusters and Management of Update (new signatures). The IDS-M is based on Intrusion Detection System (IDS) and Living Environment of Mobile Agents named Agents Agency (AA). Finally, all communication between these components is provided by mobile agents. According to the types of IDS, there are network based (NIDS) and host based (HIDS) intrusion detection systems. Then, some systems may attempt to stop an intrusion attempt but this is neither required nor expected of a monitoring system.

We discuss the functioning of our framework based on figure 4. Then, if there is probability of attacks, Static Agent sends an alert message to ID events (like ID-Events in the Figure.4) to IDS-C. After treatment of ID-Events by IDS-C, it uses Investigative Mobile Agents (IMA) for collecting evidences of attack from all the attacked VMs for further analysis and auditing. In case of attack, IDS-C aggregate malicious data, then placing them in its temporary database. Then IDS-C generates Transfer Mobile Agents (TMA) for notifying IDS-Cr placed in the cluster layer. Moreover, IDS-Cr dispatches Investigative Mobile Agents to any IDS-C those send TMA for aggregation and collection of their malicious data from the database temporarily. Then IDS-Cr uses all malicious data collected by IMA and using them to generate new signatures through a Signature Generation Algorithm (SGA).

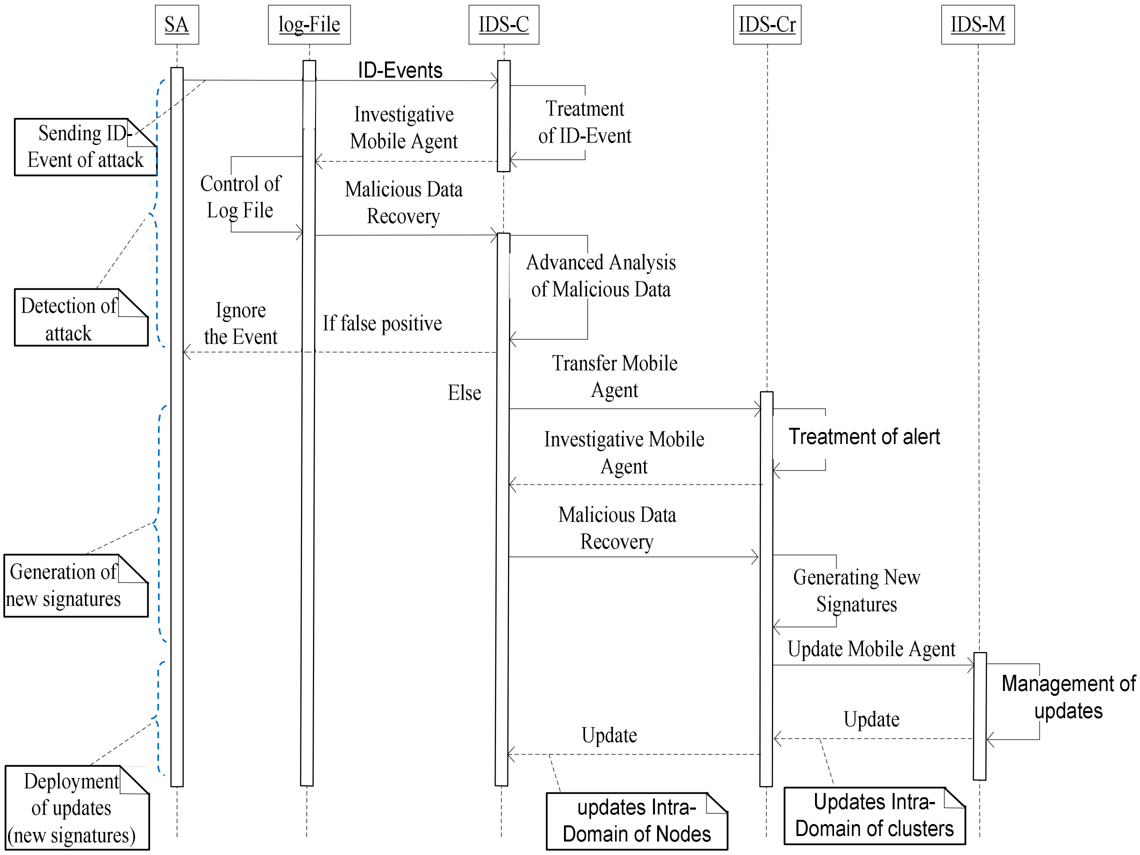

Figure 4. Principle of our framework

Finally, new signatures will be used to update the database NIDS belonging to this IDS-Cr, then the database to NIDS belonging of IDS-Cr the cluster neighboring and also their IDS-C. These updates go through the IDS-M, to maintain a hierarchical structure in our framework. Among the advantages of our approach, other clusters are protected against the same category attack. Then, our approach consists of the following features:

- Continuous detection of attacks;
- Incrementally deployable security elements;

- Dynamically enable / disable / upgrade security elements.

3.3 Analyzing the Functioning of Our Framework over Cloud Computing

3.3.1 VMs-Layer and Node-Layer for detecting attacks.

Those layers consist of four main components namely IDS Control (IDS-C), Agents Agency, Specific Static Agent Detectors (SA), and Specialized Investigative Mobile Agent (IMA).

As can be seen from VMs-Layer, Static Agents (SA) should generate an alert whenever they detect suspicious activities, then save those activities information in a log file and send alert's ID (like ID-E in the Figure.5) to Analyser Agent in IDS-C. It uses a database for analysis ID-Event, then it stores the IP address and type of attack in the file called victim host list (VHL). It sends a profile which describes the type of attack to Agent Generator. Then, agent generator retrieves the IP address from VHL and requests of Mobile Agent Dispatcher (MA-D) to generate Investigative Mobile Agents (IMA). MA-D will send IMA with a specific task, to every agency that sent similar alerts. According to Figure 5, IMA will visit and investigate all those VMs (Log-File), collect information, correlate it and finally send or carry back the result to Alerting Agent. Consequently, Alerting Agent in IDS-C will analyze the coming information and compare and match with intrusion patterns in IDS-C database [26]. Then, it will raise the alarm if it detects an intrusion. IDS-C saves the information received from IMA into its database temporary (DBT). Names and identifications of possibly discovered compromised VMs will be black listed and sent to all VMs except the black listed VMs. Then, it changes trust level manager (TLM) for the infected machines. When the Administrator finds out a new VM in the black list, the necessary actions should be taken. Those actions are quite different compared to actions against compromised physical machines. That's because virtual machines are dynamic and can be readily cloned and seamlessly moved between physical servers. That's why vulnerabilities can be unknowingly propagated. Thus, Virtual machines which labeled as compromised are recommended to be banned from migration as migration of compromised VMs may lead to propagation of intrusion. Periodically, every SA should transmit a "Hello" heart beat message to the IDS-C at regular intervals to indicate their status. In cases when these messages are not received, there is possibility of intrusion on SA or VM. Finally, Alerting Agent request generation of the Transfer Mobile Agent (TMA) for alerting IDS-Cr at level of the Cluster layer, which will be the aim of the next section.

1) Agents Agency: Agency presents an environment for mobile agents to become alive. An agency is responsible for hosting and executing Agents in parallel and provides them with environment so that they can access services, communicate with each other, and migrate to other agencies. Besides, an agency protects the underlying VMs from unauthorized access by malicious Agents.

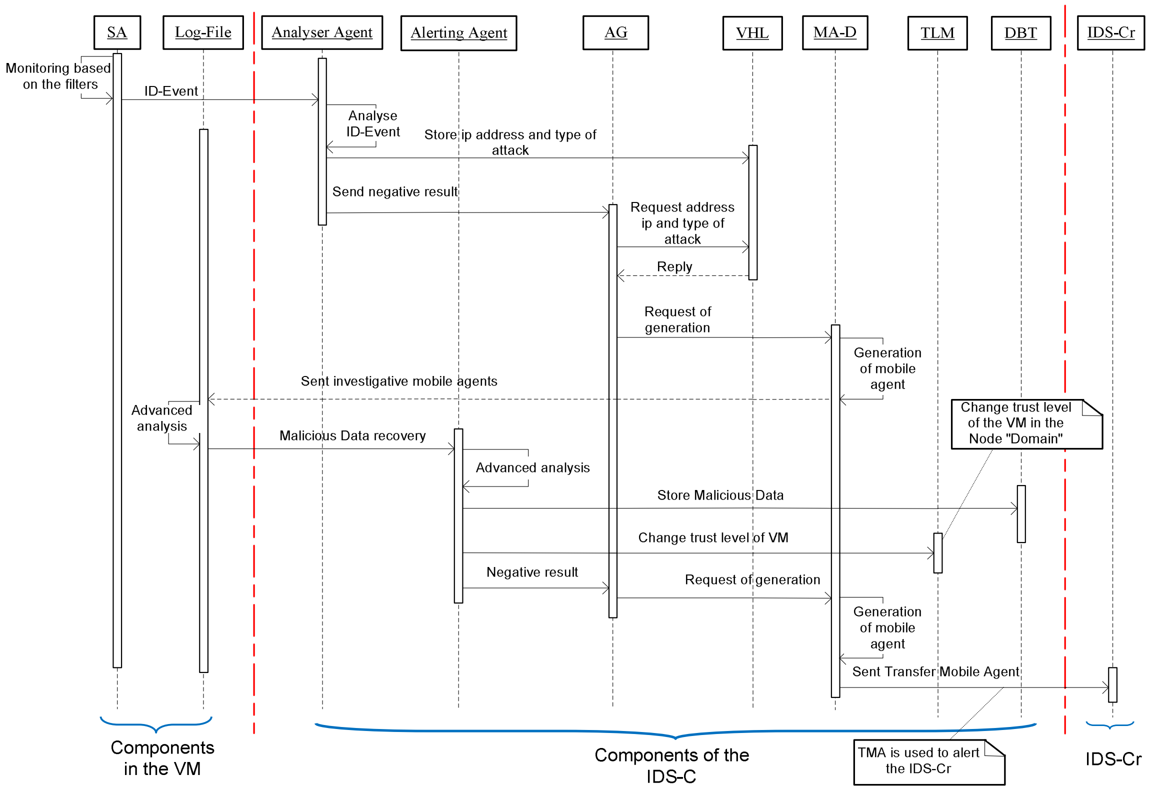

Figure 5. Process of observation and detection of attacks

2) Application Specific Static Agent: Static Agent (SA) acts like VM monitors. It generates ID events whenever traces of an attack is detected, and these events are sent in the form of structured messages to IDS-C [29]. SA is capable of monitoring the VM for different classes of attacks. The Static Agent is responsible for parsing the log files, checking for intrusion related data pattern in log files, separating data related to the attack from the rest of the data, and formatting the data as required by the investigative mobile agents (IMA). To impose the least CPU load on the VM, we use several SA with of specific tasks. Then, which means if for example a VM is hosting a mail server only, it is not logical to have one SA for monitoring all possible log files in system.

3) Investigative Mobile Agents: They are responsible for collecting evidences of an attack from all the attacked VM for further analysis and auditing. Then, they have to correlate and aggregate malicious data from all infected virtual machines, in order to detect distributed attacks. Each IMA is only responsible for detecting certain types of intrusions. This makes it easier for updating when new types of intrusion are found or new types of detection method are invented. In addition, Mobile Agents transport less data and code which save bandwidth. The IMA uses List of Compromised Agency (LCA) to identify its itinerary for visiting Hosts.

4) Intrusion Detection System Control (IDS-C): It is a central point of detection and correlation malicious data in each physical node. It includes all the components that a normal VM does and also following components:

Analyzer Agent: This agent communicates with the Data mining inference engine and its database to analyze the authentication information, to analyze the request and the response of the system at the request of SA. As a result, the analyzer agent prepares a list of information to the MA-dispatcher, after a thorough analysis of the message (ID-E) based on two databases (Behavior and knowledge).

Victim Host List (VHL): The VHL is a subcomponent of the IDS-C. The architecture is shown in Figure 5. The VHL contains lists that store the IP addresses of hosts on which suspicious activities are detected [29]. The address of the SA that generated the ID event is added to a list in the VHL through analyzer agent. The VHL maintains separate lists for each type of attack, and then VHL stored the addresses according to the type attack selected by the analyzer agent. The VHL provides the itinerary for the movement of an MA within the network for visiting the VMs infected.

Mobile Agent Dispatcher (MAD): MAD dispatches investigative Mobile Agents to the VMs based on the list sent by analyzer agent. In addition, it determines list of compromised Agencies (LCA) for IMAs.

Trust Level Manager (TLM): Defines trust level for all Nodes in the Cluster, furthermore it keeps the trust level of the other Nodes in the same neighbourhood of the cluster. There are three trust levels: 1-normal 2-suspicious 3- critical. Trust level changes based on SA and IMA investigation results. The trust level of all Nodes in the Cluster can be modified by the Trust level manager.

Agent Generator (AG): Agent generator generate task specific Agent at the request the Mobile Agent dispatcher, the agents generated (SAD and IMA) allows the detection of intrusion. Furthermore the new intrusion by using knowledge that is generated by the data mining inference engine or obtained from previous experiences. If there is a new intrusion detected, the Transfer Mobile Agent (TMA) will be generated to communicate with the upper layer.

Alerting Agent: This component compares the spotted suspicious activity with intrusions' database if they are matched, it raises the alarm and notified MA-dispatcher to generate TMA which will be transferred to the upper layer. Then the malicious data correlated and aggregated are stored in a database temporary (DB temporary in the Fig. 5). Thereafter these malicious data will be recovered by the IDS-Cr to generate new signatures.

3.3.2 Generate New Signatures in Cluster-layer

As can be seen from Cluster-Layer (Fig 2), our proposed hybrid model (IDS-Cr) in Cluster-layer each Front-end Cluster consists of three main components namely Signature Generation Algorithm, Agents Agency (living environment of mobile agents) and NIDS.

As shown in Figure 6, we combine NIDS, Signature Generation Algorithm and Mobile agents in our IDS-Cr. We use NIDS for detecting network intrusions; whereas, the signature generation algorithm is used to generate new possible signatures from partially known signatures. The NIDS and signature generation algorithm are chosen due to their following characteristics:

- **NIDS**: NIDS has the ability to perform real-time traffic analysis and packet logging. It performs protocol analysis, content searching and content matching. NIDS comprises of multiple components that communicate with each other in order to detect intrusion according to its signature database. It's configurable and constantly updated.
- **Signature Generation Algorithm (SGA)**: Different sessions of attacks are given as input to Signature Generation Algorithm (e.g, Apriori Algorithm and Signature Apriori Algorithm). According to support and confidence value rule are generated by Signature Generation Algorithm. These rules are given to NIDS. When attack is generated for which signature is stored in NIDS, it generate alarm.

After the detection of a new intrusion, the latter will be declared by IDS-C. Then IDS-C will send alert to IDS-Cr by TMA. In this case, the Analyzer agent controls the identification of TMA to protect IDS-Cr against malicious mobile agents. Then, agent generator requests of the Mobile Agent Dispatcher (MAD) to generate IMA. MAD will send IMA with a specific task, to any IDS-C that send TMA for aggregation and collection of their malicious data, based on the list sent by analyzer agent. In this case IMA performs a simple task; it is the recovery malicious data from DB temporary existing in IDS-C (shown as DB temporary in Fig. 6).

Different malicious data retrieved by the IMA, are given as input to Signature Generation algorithm (SGA). As an output, it generates new attack signatures and updates them as rules in NIDS (DB NIDS). These new signatures are used in NIDS for detecting the derivatives of known attacks. In such a way, our design can be used to detect known attacks as well as derivative attacks. It includes detecting distributed attacks (DoS/DDoS) thanks to the correlation of malicious data from multiple nodes targeted by this attack.

As shown in Figure 6, agent generator notified Mobile Agent dispatcher to prepare Mobile Agent Temporal (MAT) and Mobile Agent Update (MAU). The first for notified Cluster Administrator by generating a new signature and at the same time the second mobile agent transporting new signatures to

IDS-M. This is the crucial part of our approach, the new signatures will be sent to existing IDS-M in the Front-end Cloud. In the next session, we discuss the advantages of this part.

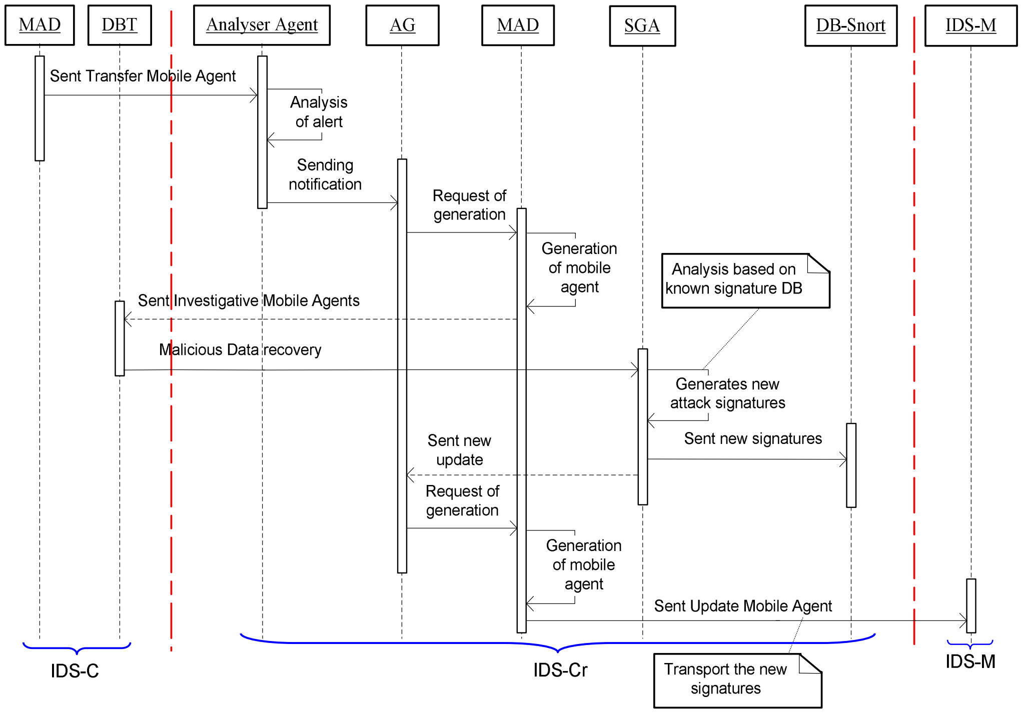

Figure 6. Using malicious data for generate new signatures

3.3.3 Deployment of Update by Cloud- layer

As shown in Figure 7, After detecting a new attack in a Cluster, the IDS-Cr will produce new signatures, which will be used to protect the Cluster attempts to tell the same attack. For this reason the IDS-M will recover the new signature with MAU then update its database (DBM) and subsequently sent the MAU to neighboring IDS-Cr for update their database, with the exception of the one who sent the new signature.

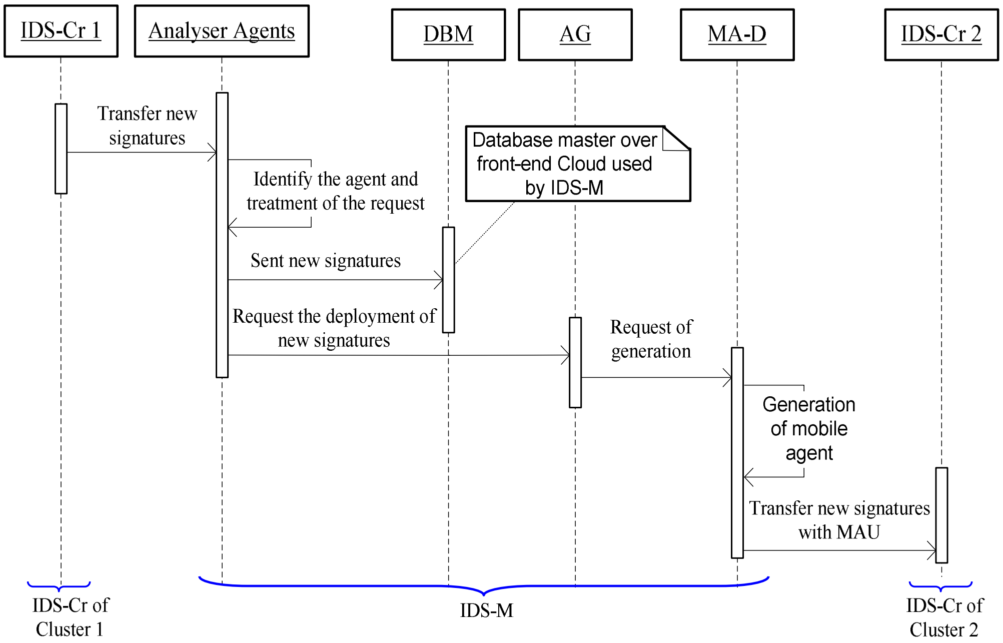
IDS-Cr 1
Analyser Agents
DBM
AG
MA-D
IDS-Cr 2
Transfer new signatures
Identify the agent and treatment of the request
Database master over front-end Cloud used by IDS-M
Sent new signatures
Request the deployment of new signatures
Request of generation
Generation of mobile agent
Transfer new signatures with MAU
IDS-Cr of Cluster 1
IDS-M
IDS-Cr of Cluster 2

Figure 7. Deployment of new signatures between different clusters

4. Discussion

Cloud computing is an internet based computing technology, in which required resources are provided in rented basis to customers. Then, existence of vulnerabilities in Cloud computing allow intruders to affect the confidentiality, availability and integrity of cloud resources as well as services. Detection intrusion and other network level malicious activities are major security concerns in the Cloud. For ensure a high level of trust in cloud computing, we propose a new framework based on cooperative of Hy-IDS and mobile agents. This framework, allowed us to achieve three objectives, namely: intrusion detection (known attacks as well as derivative of known attacks) at the front-end as well as the back-end of Cloud environment (i.e IaaS) of manner autonomous. Then, generating new signatures from malicious data. Finally, dynamic deployment of updates between clusters in a cloud computing, using the newest signatures previously created. We used the signature generation algorithm and exchange of updates between clusters to achieve new knowledge and detect new kind of intrusion. Outstanding scalability is another strong point for this framework. When for example our VM migrates from server machine to another one (e.g. from Cluster-1 to Cluster-2), it is still possible to perform intrusion detection as our IMA can migrate just like VMs, and the same rule applies to other mobile agents (Transfer Mobile Agents and Mobile Agent Update). And this is the strength of our framework which gives the IDS and NIDS great scalability and flexibility. Therefore, we have met almost all the mentioned challenges in our framework. Therefore, this framework has several advantages for this reason it can be considered as an effective solution for the detection of intrusion into cloud computing. Thus, it can be used to protect people and property against risks of intrusion and aggression.

5. Conclusions and Future Works

Cloud Computing is undergoing an incontestable success, which could be indeed compromised by concerns about the risks related to potential misuse of this model aimed at conducting illegal activities. There is a major need of bringing security, transparency and reliability in cloud model for client satisfaction. Then, one of the security issues is how to reduce the impact of any type of intrusion in this environment. Thus in this paper, we propose an intelligent framework, which is based on the collaboration of the IDS-C, IDS-Cr, IDS-M and Mobile agents. As mentioned previously, mobile agents are used in our framework to investigate the VMs, transfer of malicious data, and exchange of update between different clusters in cloud computing; so that the mobile agents could have the ability to investigate the VMs and ensure communication between hierarchical layers or cluster. They should be granted a permission of access the host's resources like file system, network interfaces, database and so on. There are two options; first is to give them every right to access all resources. Second is to restrict their access to the resources which they need for investigation [30]. Therefore, IDS-C and IDS-C should issue a certificate which will authorize a mobile agent to access to certain resources on remote Hosts (e.g. for communication between the IDS-C and VMs or between IDS-C and IDS-Cr). However, it is mentioned above that IDS-C, IDS-Cr and IDS-M which use mobile agents take over the beginners toolboxes of the mobile agents such as security architecture flaws. Consequently, further development of mobile agent's toolboxes will make it easier to apply them into IDS systems.

The challenges mentioned in this paper are the following: the first, intrusions detection in a virtual environment using mobile agents in order to collect, transfer malicious data. The second, generation of new signatures from malicious data. Finally, dynamic deployment of updates between clusters in a cloud computing, using the newest signatures.

Further research can be undertaken to improve the work presented. The future directives are:

- We will continue to deepen the concepts and the notions of this architecture and to proceed after to its implementation in order to validate it.
- Take into account the adaptability of agents' appearance.
- Use of cooperation mechanisms between mobile agents in order to effectively perform the tasks required.
- Generation of response actions (local and remote).

References

[1] Mohemed Almorsy, et al, "Collaboration-Based Cloud Computing Security Management Framework". IEEE 4th International Conference on Cloud Computing, 2011.

[2] H. TOUMI, A. EDDAOUI and M. TALEA." Cooperative Intrusion Detection System Framework Using Mobile Agents for Cloud Computing". Journal of Theoretical and Applied Information Technology 10th December 2014. Vol.70 No.1

[3] Sebastian Roschke, et al. "Intrusion Detection in the Cloud". Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, 2009.

[4] N. Jeyanthi, N.Ch.S.N. Iyengar, "Packet Resonance Strategy: A Spoof Attack Detection and Prevention Mechanism in Cloud Computing Environment", International Journal of Communication Networks and Information Security (IJCNIS) Vol. 4, No. 3, December 2012.

[5] M. Armbrust, A. et al, "Above the Clouds: A Berkeley View of Cloud Computing", UC Berkeley Reliable Adaptive Distributed Systems Laboratory, February 10, 2009.

[6] Saeed M. Alqahtani, et al. "An Intelligent Intrusion Prevention System for Cloud Computing (SIPSCC)". International Conference on Computational Science and Computational Intelligence, 2014.

[7] K. Benzidane, et al. "Autonomous Agent-based Inspection for inter-VM Traffic in a Cloud Environment". The 7th International Conference for Internet Technology and Secured Transactions, 2012.

[8] Suruchee V.Nandgaonkar, et al. "A Comprehensive Study on Cloud Computing". International Journal of Computer Science and Mobile Computing, April- 2014.

[9] Agarwal, A.; et al. "Reviewing the world of virtualization". Intelligent systems modeling and simulation (ISMS), Third international conference, 2012, PP.554 -557.

[10] A. Elsayed, N. Abdelbaki. "Performance Evaluation and Comparison of the Top Market Virtualization Hypervisors". 8th International Conference on Computer Engineering & Systems (ICCES) – 2013.

[11] Lei Yu, Chuliang Weng, Minglu Li, and Yuan Luo, "An Efficient Para-Virtualization Snapshot Mechanism for Virtual Disks in Private Clouds", IEEE Network July/August 2011.

[12] Anish Babu S, et al. "System Performance evaluation of Para virtualization, Container virtualization and Full virtualization using Xen, OpenVZ and XenServer". Fourth International Conference on Advances in Computing and Communications. 2014 IEEE.

[13] Miguel G. Xavier, Marcelo V. Neves, Fabio D. Rossi, Tiago C. Ferreto, Timoteo Lange, Cesar A. F. De Rose, “Performance Evaluation of Container-based Virtualization for High Performance Computing Environments”, 21st Euromicro International Conference on Parallel, Distributed, and Network-Based Processing IEEE 2013.

[14] Michael Terrell, Natarajan Meghanathan, “Setting up of a Cloud Cyber Infrastrcture using Xen Hypervisor”, 10th International Conference on Information Technology: New Generations IEEE 2013.

[15] Fu Wen & Li Xiang, “The study on data security in Cloud Computing b ased on Virtualization”. In II. In Medicine and Education (lTME), International Symposium on. pp. 257-261, 2011.

[16] Suryanarayana, v., Jasti, A. & Pendse, R., "Credit scheduling and pre fetching in hypervisors using Hidden Mark ov Models”, In Local Computer Network s (LCN), IEEE 35th Conference on. Local Computer Network s (LCN), IEEE 35th Conference on. pp. 224-227, 2010.

[17] S. Nawaz Brohi, et al. "Identifying and Analyzing Security Threats to Virtualized Cloud Computing Infrastructures”, International of Cloud Computing, Technologies, Applications & Management. 978-1-4673-4416-6. P_155. 2012.

[18] Y. Singh, et al.” DIMENSIONS AND ISSUES OF MOBILE AGENT TECHNOLOGY”. International Journal of Artificial Intelligence & Applications (IJAIA), Vol.3, No.5, September 2012.

[19] P. Akhilandeswari, et al,” Proceedings of International Conference on Internet Computing and Information Communications: ICICIC Global”. Advances in Intelligent Systems and Computing, 2014.

[20] Josenilson Dias Araújo et al, “EICIDS-Elastic and Internal Cloud-based Detection System,” International Journal of Communication Networks and Information Security (IJCNIS), Vol. 7, No. 1, April 2015.

[21] Hanieh Jalali , Ahmad Baraani, “Process Aware Host-based Intrusion Detection Model”, International Journal of Communication Networks and Information Security (IJCNIS) Vol. 4, No. 2, August 2012

[22] Hassen Mohammed Alsafi, Wafaa Mustafa Abduallah and Al-Sakib Khan Pathan,”IDPS: an integrated intrusion handling model for cloud computing environment”. International Journal of Computing & Information Technology (IJCIT), vol. 4, no 1, p. 1-16, 2012.

[23] I. Gul and M. Hussain, “Distributed Cloud Intrusion Detection Model”, International Journal of Advanced Science and Technology, vol. 34, pp. 71-82, 2011.

[24] N. B. Dhurpate and L.M.R.J. Lobo, "Network Intrusion Detection Evading System using Frequent Pattern Matching". International Journal of Engineering Trends and Technology (IJETT) – Volume 4 Issue 8- August 2013.

[25] Chirag N. Modi, Dhiren R. Patel, Avi Patel, Muttukrishnan Rajarajan, "Integrating Signature Apriori based Network Intrusion Detection System (NIDS) in Cloud Computing". 2nd International Conference on Communication, Computing & Security, 905 – 912, 2012.

[26] Dastjerdi, Amir Vahid, Kamalrulnizam Abu Bakar & Sayed Gholam Hassan Tabatabaei. “Distributed Intrusion Detection in Clouds Using Mobile Agents”, In Proceedings of the 2009 Third International Conference on Advanced Engineering Computing and Applications in Sciences. ADVCOMP ’09 pp. 175–180, 2009

[27] P. Singh Hada, et al.” Security Agents: A Mobile Agent based Trust Model for Cloud Computing”. International Journal of Computer Applications, Volume 36– No.12, December 2011

[28] Sebastian Roschke, Feng Cheng, Christoph Meinel.” Intrusion Detection in the Cloud”. Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, 2009.

[29] Pradeep Kannadiga and Mohammad Zulkernine School of Computing Queen’s University, Kingston Ontario, Canada K7L 3N, “DIDMA: A Distributed Intrusion Detection System Using Mobile Agents”, IEEE, 2005.

[30] Amir Vahid Dastjerdi, and Kamalrulnizam Abu Bakar. “A Novel Hybrid Mobile Agent Based Distributed Intrusion Detection System”, International Journal of Computer, Information Science and Engineering Vol: 2 No: 9, 2008.