Asian Journal of Information Technology 15 (18): 3461-3473, 2016
ISSN: 1682-3915

Neuro Fuzzy ART Based User Behaviour Trust in Cloud Computing

[1]M. Jaiganesh, [2]S. Mercy, [2]K.C. Anupama and [1]A. Vincent Antony Kumar
[1]Department of Information Technology, PSNA College of Engineering and Technology, Dindigul, 624 622 Tamil Nadu, India
[2]Bangalore Institute of Technology, V.V. Puram, 560004 Bangalore, India

Abstract: Cloud computing dissemination is exposed by unsolved protection concerns that influence both the cloud provider and the cloud user. In especially, a resource based side channel attacks survived to exploit the properties of hardware and disc space, called virtual isolation problem. We present a major three factors such that Memory, Giga Floating-point Operations Per Second (GLOPS) and Disc space. To overcome this problem, we propose and examine resources based user behaviour trust for cloud environment using Neuro fuzzy and fuzzy Adaptive Resonance Theory (ART) classifications. Fuzzy ART categorizes the given cloud resource factors using unsupervised learning with vigilance threshold output to classify the hackers as namely secure, vulnerable, modified and anomaly. The next stage, neuro fuzzy method is to control the hackers by predicting the trust rate of each client. Neuro fuzzy adoption are predetermined both modified and anomaly hackers. The results are specified and compared with the trust rate of high distinction. The research hereby shows the valuable outcome to cloud service provider such that once the trust rate of the client is predicted to generate remedy actions like: snapshot of previous connections, back up awareness and accounting of trust.

Key words: Cloud computing, hypervisor monitoring, neuro fuzzy, Fuzzy ART, optimization, virtualization

INTRODUCTION

Cloud computing has become an increasingly popular enterprise model where the resources are available on-demand to the user as and when they needed. It facilitates to access the contents across the internet independently without a reference to the underlying hosting infrastructure (Buyya *et al*., 2008). Cloud service provider provides services to users and manages the entire cloud. They are assigned to design the services based on scalable applications (Keskin *et al*., 2010). It establishes a new data centers for hosting cloud computing applications in various locations around the world to provide redundancy and ensure reliability. The data center includes the resources where they are highly centralized and trusted. It provides more services and covers the maximum number of users (Jaiganesh and Kumar, 2012, 2013). So, the cloud service providers must be prepared in better tolerance to manage and update the data centers. The main goal of a data center is to provide access to a potentially vast amount of computing resources in an easy and user centric way (Dutreilh *et al*., 2010). The resources are shared and managed through the virtualization technology employed in cloud computing environment. It uses hypervisor within cluster environment that allows multiple virtual machines to share the resources allocated to them. The virtual clients must ensure that the data they receive along with applications and services are secure (Scarfone, 2011; Rong *et al*., 2013). The clients have unauthorized access to the resources(Memory, Disk space, etc) of their neighborhood and these vulnerabilities in turn make the platform more vulnerable to the attacks. As the amount of web based disseminated system rises, the occurrences of malware behavior also gets increased. So, the cloud users can execute their services by a centralized repository called resource handler where each client has their own identity (Lombardi and Di Pietro, 2011). This identity is performed multiple times remotely and falsifies the cause of Sybil attack. Hence, the hackers could utilize other resources like processor elements and also possibly utilize entire memory by creating extreme number of mail messages, inserting files in the File Transfer Protocol (FTP) area. To overcome the above behaviors in the cloud. The trustworthy of clients is considered to be major issue in cloud. Trust is always made only if sufficient services and expectation is attained. The challenges of trusting the cloud don't lie entirely in the technology; it also involves customer confidence that stems from loss of control over data assets. Though cloud computing is designed to provide better utilization of resources, the user behavior trust is

Corresponding Author: S. Mercy, Bangalore Institute of Technology, V.V. Puram, 560004 Bangalore, India

essential. The user behavior trust includes the maintenance of software without malware, subjective of virtual client to hack the others, damage of infrastructure. No matter what causes the user to mistrust yet the cloud service provider must to monitor user behavior in order to ensure the credibility of the user's identity and behavior. To manage these uncertainty problems in cloud, two techniques are proposed. The first technique is an unsupervised learning technique Fuzzy ART to classify the virtual clients based on their behavior. The second technique is Neuro Fuzzy with mamdani approach to evaluate the trust of the every individual based on their usage of the resources.

Attacks and threads: The cloud service can share its infrastructure which scales to provide a dynamic, on-demand infrastructure at a suitable cost. The provider does this by utilizing virtualization where virtual machines from multiple customers share a same physical server (Armbrust *et al*., 2009; Lin and Lin, 1997). This multi-tenancy allows mutual customers to hire resources from the same provider with the need for a secure virtualization solution. Because of its central role, the virtualization software forms a main target for the attack. In some cases the attackers exploit the vulnerabilities to the cloud and use its resource to deploy attacks in Memory, Disk space and Giga Floating-point Operations Per Second (GLOPS). Exploiting such an attack, the attacker owns the ability to access the other virtual machines and therefore reduces confidentiality, integrity and availability of the other virtual machines' code or data (Wang *et al*., 2011).

Memory attack: The main memory attack must have a physical access before the system is about to shut down. But in non-volatile memory this precondition is unnecessary. The Non-Volatile Main Memory (NVMM) introduces new vulnerabilities-sensitive data that can be extracted or modified by some client who gains access to the memory while the computer is not turned on or after the reboot. Here no freezing is required and the memory chips can be retrieved at any time. This approach mitigates the vulnerabilities of persistent main memory while retaining the advantages of non-volatile memory (Iosup *et al*., 2011; Shea and Liu, 2012). The vulnerabilities introduced by the use of NVMM lead to informal design such as the physical attacks on the suspended memory. This leads the operating system state to be easily retrievable even after shut down or reboot. So protections must be maintained without support from or trusting the operating systems running on the host. The solution to these problems is to induce a little overhead on memory access. But, the systems that expose sensitive data on disk will still be vulnerable.

Disk space attack: Outsourcing of data storage increases the attack on the surface area. When data is distributed, it is stored at more locations increasing the risk of unauthorized physical access to the data. So that the data is replicated and it depends on the service level of a customer chooses and on the service provided. Risk of unauthorized access to data can be mitigated through the use of encryption which can be applied to data as part of the storage service (Wang *et al*., 2010). Sometimes because of error in actions that takes place, a bug, etc., the risk applies to entire cloud storage Encryption techniques protects the data as it is being transmitted to and from the cloud service provider.

CPU attack: It is observed that the utility computing server could cheat in CPU time, as the entire platform is under the server's control. CPU time is different from response time or turnaround time which refers to the elapsed time between user's job submission and the result output (Luo *et al*., 2008). Compared with CPU time consumption, the turnaround time is less stable, since it significantly depends on many other factors, such as the system load or the scheduling policy. Therefore, the turnaround time does not truly reflect the amount of resources consumed by a user's job. A CPU is trustworthy if and only if the measured time equals to the outcome from the same job executed in the user's own platform with the same hardware/software specification. Sometimes a dishonest service provider has a strong motivation to inflate the CPU time of a user job by various attacks such as shell attack, process scheduling attack, etc.

Literature review: (Dan J. Kim) an appropriate natural world of internet services in cloud computing where clients would face a huge level of risk. I sprit, the clients construct a gamble about the uncertainty of the expectations. In these uncertain conditions, when clients have to respond, the trust arrives for the clarification of the difficulty of risk. Trust suits the critical policy for treating with a vague environment. Mohammed Alhamad state that the trust is a vital factor and having different dimensions in nature. The aspect is overcoming the protection threats and malicious hits. The dimensions such as names scalability, availability, security, usability factors of the cloud services. The above said factors are further into clustering by use of fuzzy aspects and linguistic terms. The trust value of cloud service provider

is calculated by utilizing the sugeno fuzzy inference approach (Worku *et al*., 2014). Sheikh Mahbub Habib classified trust as a soft and hard. They proposed several featured trust management architecture for the business approaches. It comprises the trust worthy and other following attributes: compliance, data governance and information security. They developed a Consensus Assessment Initiative Questionnaire (CAIQ) based trust information using the Cloud Control Assessment (CCA) tool (Habib *et al*., 2014). Kuehnhausen *et al*. (2012) presented a measurement of the resources called armor. It is a framework used to attest the estimated or exceeded use of the cloud resources and their behaviours. For each attestation, a well set of data and subsequent data of all resources and the contexts are gathered. Based on the measurement, e-Resources are negotiated based on the assessment and confidence intervals. Patel *et al*. (2013) mentioned that the nature of cloud computing was pertinent to the resilient competences. It refers to an on-demand computing to modify the size of resources based on the number service requests. This agreement will require variety of resources with different quantities. The resources should be securely distributed based on the requirements. In cloud computing distribute their services in a linearly incremental way by sharing the resource infrastructure. In order to provide isolation attributes for super scalar architecture. A hypervisor acts as a transformer between the virtual client and the data center computing resources. In this situation, a hypervisor face lot of defects in controlling action and resource provisions. So, it should recommend by providing security measures and monitoring. Jingwei Huang recommended a trust as in two dimensions. Those are policy based and evidence based on the trust prediction. Evidences is an attribute based method for the trust judgments. It includes the specific properties of a cloud service taken as an evidence and the faith in those based on the formal guarantee and continuity of trust based on justification. A trust evaluated from the behaviour can reproduce trust worthiness (Huang and Nicol, 2013). Ibrahim predicted a dynamic neural network based on the intrusion detection systems. It is used to classify the data as normal or vulnerable. The unsupervised neural network provides a unsupervised learning neural nets to separate user behaviour as a patient or an intrusive one. The system incorporates the tuning, testing and training of distributed time delay neural network in the intrusion detection systems (Ibrahim, 2010).

Yashashree offered that a trust is a formless in nature. Indistinct is the famous intrinsic nature in the trust computing fuzzy logic is also a multi-valued logic used to find the trust of the user behaviour. Many number of fuzzy models have been proposed and fuzzy rules to predict the vague in trust management. The fuzzy control system is always used in control problems and decision making systems (Bendale and Shah, 2013).

Problem statement: The underlying problem of a cloud service for virtual clients is demonstrated in Fig. 1. Three elements of the architecture can be spotted as follows:

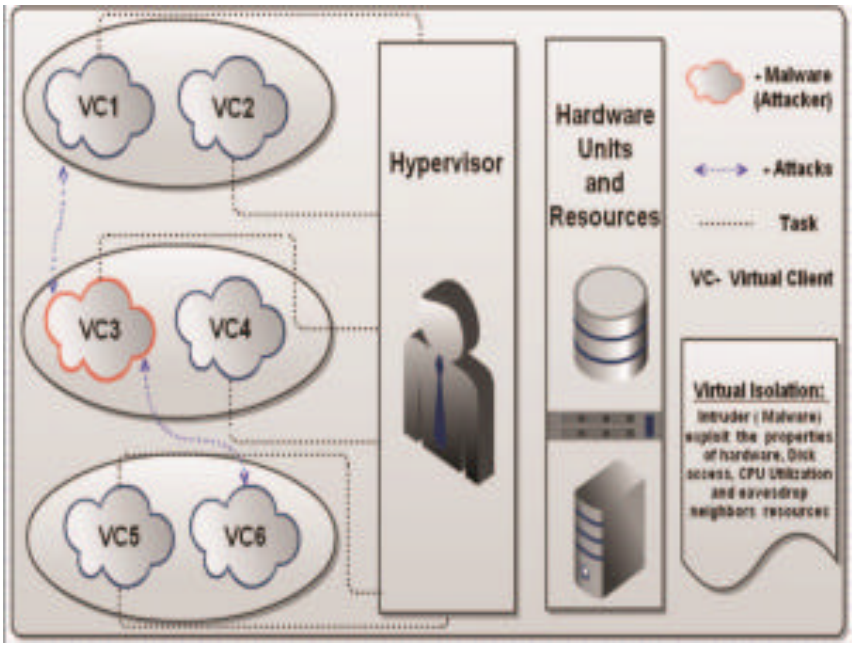

Fig. 1: Virtual isolation problem

- Data center: It is a sophisticated server having high volume of disk capacity, hardware units and resources. It is shared by the virtual clients for storage and availability
- Hypervisor: It is a virtual machine monitor that is responsible for sharing the data center resources between the virtual clients
- Virtual client: The user request for a service from the data center to the virtual client. The resource in the data center is utilized by the virtual clients. The service requested can be Software as a service, platform as a service and Infrastructure as a service

The virtual clients requesting for a service, suffer from the problem line sharing the resources and availability. It is solved virtually with the help of hypervisors available. The virtual clients request for a service from the cloud (data center) by making hyper calls to the hypervisor. The sharing of resources is done by the hypervisor but the most imperative circumstance to be noted is the virtual hypervisor security. The major issue in the hypervisor security is the virtual isolation problem. It is depicted in Fig. 1. In this Fig. 1, there are six virtual clients namely VC1, VC6 using their services. The virtual client VC3 is a malware that is able to attack the virtual client VC4 and VC6. The resources of the legal virtual clients are hacked by the malware injected into the virtual client and it starts gaining its complete control over the hypervisor later.

Goals: The goals and benefits of the proposed system is listed.

Snapshot of previous connections-establishments: The snapshot is a documentation of entity. It is an entity which updates the previous state and the current state to build an image. In case of any abnormal situation, hypervisor initiates snapshot action as cyclic process. These files are automatically stored in a permanent memory and is redundant in other systems also. These files contain updating of virtual memory, processor, disk space and network information.

Backup awareness: Once the hypervisor suspects an attacker, it safeguards the entire resources within them, because the data center accessed by hypervisor is a pool of data to be stored in it. The concept of restore action is initiated by the hypervisor and it exchanges the entire data as backup into another sophisticated systems.

Accounting and trust: In cloud computing not only single client is accessing but a millions of users are utilizing the service. So, to notify an attacker, the hypervisor should be able to find the entire clients transaction and account information's to be updated, because, the service utilized in cloud computing are should be paid by use only. Once a client is identified, he has to be put into hackers lists and zero for their trust. It is one of the behavior trust predictions in nature.

Problem formulation: The virtualization in the guest Operating System (OS) on a host are managed by the hypervisor. It usually controls the instruction flow that takes place between guest OS's and also physical hardware such as memory, disk space and the GFLOPS. Resources are partitioned by hypervisor either physically or logically. While physically partitioned hypervisor assigns a separate physical resources to each guest OS such as Disk space. While it is logically partitioned it allows multiple guest resources to share the same physical resources such as Random Access Memory (RAM) and processors. The hypervisor function is to partition the system's resources and to isolate the guest OS's so that, each can access only the resources allocated to them. But at times they can share the shared resources as files on the host OS. The hypervisor is capable to monitor the trust of the each guest OS running within it. The most important issue to be considered is that the hypervisor should be carefully monitored for signs of compromise (Modi *et al*., 2013). By its functionality, it prevents unauthorized access to resources and also helps to prevent one OS from the other, i.e., the malware guest OS cannot inject the insecure files into another guest OS's memory. The main aim to isolate the guest OS's from one another and hypervisor is to be aware of side channel attacks. These attacks can exploit the physical properties of hardware to reveal information about the usage of amount of memory access, CPU and other resources. Attackers try to break out of a guest OS so that they easily access the other guest OS and hypervisor too. Sometimes the attacker compromise the hypervisor and gain the control over its entire guest OS's. So, it is important to secure each hypervisor by some security policies. The hypervisor must be designed in some concern that they cannot be detected by the attackers. By some additional layer of security, the attacks are prevented against hypervisor detection technique which includes checking of file system, memory and registry. The attackers presence are incurred.

Giga Floating-point Operations Per Second (GLOPS): GFLOPS or GigaFLOPS, measures a quantity in billions of Floating-point Operations Per Second (FLOPS) that a computer's microprocessor can handle. Frequently,

the word Giga FLOPS is perplexed with frequency (Jaiganesh and Kumar, 2013). The difference is that frequency measures the number of cycles the CPU runs at and the GFLOPS calculates the number of floating point operations it can handle. Thus, FLOPS is a standard that shows how the system executes while computing very difficult mathematical estimations. If GFLOPS are high, it automatically indicates that the usage of the data center is enormous and has a chance for an intruder attack. Thus, it is considered to be an important factor in determining the trust of the cloud.

Memory: It stores applications for fast retrieval. The memory should be elastic in nature so that the applications are performed. It purely depends on application or task used by the client (The HPCC Team in 2009). The applications and the files are permanently stored in data center by accessing third party clients and users. Example: Amazon's Simple Storage Service (S3). Many CPU's transactions are done in a single data center. So, memory is inversely able to tolerate CPU transactions and trust evaluations (Jaiganesh and Sivasankari, 2012).

Disk space: Disk space is the storage capacity available for a particular virtual client. It purely depends on the application or task used by the client (Jaiganesh and Sivasankari, 2012; Xue *et al*., 2012). In cloud computing, the applications are permanently stored in the data center for the access of the users whenever they need. Thus the memory should be elastic in nature to support the changing applications of all the users. In the case of SaaS (Software as Service), the memory usage is small where as in the other two services, it is comparatively high.

Preliminaries: Definition 1-Adaptive Resonance Theory (ART): The ART1 Architecture consists of a layer of linear units where training iteration consists of taking a training example and examining with the existing prototypes (Carpenter *et al*., 1991). It uses comparison layer to receive the input but comprises of single inputs whereas in case of ART2 it comprises multiple capabilities to support the continuous inputs. The ART2-A extends with drastically accelerated runtime and produce qualitative results.

Definition 2 (fuzzy implication): In general, fuzzy implications ∂ is defined as the function of the form:

$$\partial: [0, 1]\times[0, 1]\rightarrow[0, 1] \quad (1)$$

It gives any of possible true values a, b of given fuzzy proposition p, q, respectively define the true value, ∂(a, b) of the conditional proposition called if then rules like "if p, then q". This is called classical implication of $p\rightarrow q$ from the restricted domain {0, 1} to the full domain [0, 1] of true values in fuzzy logic deriving '∂' in classical formula is:

$$\partial(a, b) = \bar{a}\vee b \quad (2)$$

for all $a, b\in\{0, 1\}$. We interpret disjunction and negation as a fuzzy union and fuzzy complement then, ∂ in classical logic is to employ the equation:

$$\partial(a, b) = \max \{x\in \{0, 1\}|a\wedge x\leq b\} \quad (3)$$

Due to law of absorption of negation in classical logic as either:

$$\partial(a, b) = \bar{a}\vee(a\wedge b) \quad (4)$$

Definition 3 (fuzzy proposition): The proposition are measured in their ranges and true values. It depends on the matter of degree. So, each fuzzy proposition is uttered by a number in the element interval [0, 1]. We consider our model as conditional and unqualified propositions, Propositions 'P' of this type are expressed by the canonical form:

$$P: \text{If } x = A, \text{ then } y = B \quad (5)$$

where, x and y are variables whose values are in set X and Y, respectively. Finally A and B are fuzzy sets on X and Y, respectively. The propositions may also be viewed as:

$$\langle x, y\rangle \text{ is } R \quad (6)$$

where, R is a fuzzy set on $X\times Y$ that is determined for each $x\in X$, $y\in Y$ by equation:

$$R(x, y) = \partial[A(x), B(y)] \quad (7)$$

where ∂ denotes a binary operation on [0, 1] representing a suitable fuzzy implication.

Definition 4 (compositional rule inference): Consider variables x and y that takes values from sets X and Y, respectively and assume that for all $x\in X$ and all $y\in Y$ the variables are related by a function $y = f(x)$ and x is in a given set A and y is in a given set B is given by:

$$B = \{y\in Y|\langle x, y\rangle\in R\} \quad (8)$$

Similarly that $x\in A$, we can infer $y\in B$:

$$B = \{y\in Y|\langle x, y\rangle\in R, x\in A\} \quad (9)$$

Examine that this inference may be expressed equally well in terms of characteristics functions X_A, X_B, X_R of sets A, B, R, respectively by the equation:

$$X_B(y) = \sup_{x \in X} [X_A(x), X_B(x, y)] \quad (10)$$

for all $y \in Y$. Let us proceed now one step further and assume that R is fuzzy relation on $X \times Y$ and $\bar{A}, \bar{B}$ are fuzzy sets on X and Y, respectively. Then if R and $\bar{A}$ are given, we can obtain $\bar{B}$ by the equation:

$$\bar{B} = \sup_{x \in X} [\bar{A}(x), R(x, y)] \quad (11)$$

for all $y \in Y$ which is the generalization of Eq. 6 obtained by replacing the characteristics functions in Eq. 6 with corresponding membership functions. We prefer this equation as generalization called compositional rule of inference to facilitate approximate reasoning.

Theorem-1: In Fuzzy ART architecture, all the templates are distinct. Assume that Fuzzy ART creates two templates W1 and W2 are equal. W1 is created first and template W2 is created by the input pattern coded by template W2.

$$W2 = \beta(I \wedge W2') + (1-\beta)W2' = W1 \quad (12)$$

Theorem-2: In a Fuzzy ART architecture with sufficient number of nodes in the F2 layer, the size of a template is larger than $\alpha M/(\alpha+M)$:

$$|W| > \alpha M/(\alpha+M) \quad (13)$$

Theorem-3: In a Fuzzy ART architecture, if a node J in the F2 layer has perfectly learned an input pattern I, then when I is presented it will directly access node J:

$$T_j = |I \wedge W_j|/(\alpha+W_j) \quad (14)$$

MATERIALS AND METHODS

Proposed method: The underlying architecture of the proposed method is depicted in Fig. 2. The System is composed of several elements such as hypervisor, virtual clients, virtual service provider etc. In cloud computing environment, multiple instances of a variety of operating systems may share the virtualized hardware resources. Hypervisors are an important component of virtualized environments. Hypervisors are programs that allow multiple operating systems, known as guests to run in virtual machines in an isolated fashion and thus share a single physical machine or host. It is responsible for

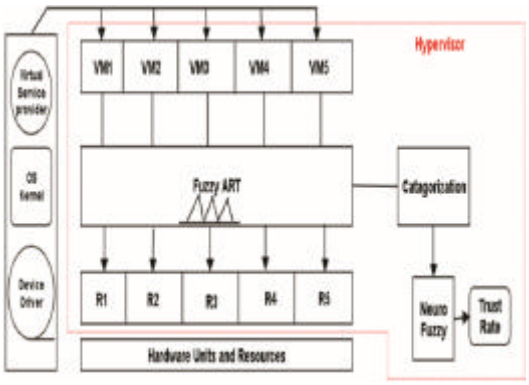

Fig. 2: System architecture

sharing the data center resources between virtual clients. The virtual clients request for their service from the cloud (data center) by making hyper calls to the hypervisor and utilize the resources. While the virtual client request for their service, it suffer from the problem of sharing the resources available. It is solved virtually with the help of hypervisor available.

The sharing of resources is done by the hypervisor but the most imperative circumstance to be noted down is the virtual hypervisor security. Since Hypervisor based cloud servers are always exposed to attacks, it can easily be exploited to take down the whole cloud along with its resources. In this Fig. 2, each client in the cloud utilizes it resources allocated by the data center. When the targeted virtual client is attacked by the attacker, the resources will be hacked by the malware injected virtual client and it utilizes the resources of other virtual clients. The active interaction time of the hypervisor is minimized by allocating the resources, I/O calls before the start of the collaboration. Even though hypervisor interaction is reduced, the malware injected virtual client starts gaining its control over the hypervisor and it may also get attacked sometimes. To illustrate this problem the proposed system uses Fuzzy ART and Neuro Fuzzy Techniques (Blume and Esener, 1997). This strengthens the hypervisor to find out the trust in the usage of resources by the virtual clients. The fuzzy Adaptive Resonance Theory (ART) technique uses the input factors such as Memory, GFLOPS, Disk Space for each virtual client and follows an unsupervised learning method to train and test the virtual clients (Wang *et al*., 2010). With this fast learning algorithm, the virtual clients are classified into four categories such as Secure, Vulnerable, Modified and Anomaly based on the vigilance threshold. Smaller the vigilance threshold higher categories are obtained. The output classes thus inferred modelled with neuro fuzzy system which uses fuzzy

Inference rules. The fuzzy inference system takes input factors such as memory, GFLOPS and disk space for functional operations such as fuzzification (Tian *et al*., 2010), fuzzy inference rule and defuzzification from which the trust of the virtual client is factored out. The virtualization problem where the resources hacked by the attacker and the trust of the individual client is solved by using the Fuzzy ART and Neuro Fuzy system. The trust rate determined based on these techniques is analyzed and compared with the existing approaches (Huang *et al*., 1995).

Categorization of attacks using Fuzzy ART: In ART architecture, if an input pattern is different from the patterns (i.e., weights) that have stored in the network, then it will create new category and different input pattern will be associated with it (Nagamwitthayanon and Wattanapongsakorn, 2011). The ART1 (by Definition 1) was designed to process binary data only whereas the proposed Fuzzy ART is designed to learn both the binary and analog patterns (Frank *et al*., 1998). The Fuzzy ART uses unsupervised learning category. This Network consists of two layers, the input layer F1 (Resources: Memory, GFLOPS, Disk space) for each virtual client in the network and the output layer F2 (categories: Secure, Vulnerable, Modified, Anomaly). The F1 neuron i and the F2 neuron j are interconnected by top-down-weight and bottom-up-weight W_{ij} where W_{ij} is chosen as discussed in Theorem 1.On the other hand, the orienting subsystem has a classification precision called vigilance parameter (ρ). The learning Fuzzy ART Algorithm is as follows.

Step 1: The input resources (Memory, CPU, Disk space) of each client is normalized as $I\epsilon[0, 1]^n$ is given to input layer F1 satisfying the Theorem 2:

$$NI_{i,j} = I_{i,j}\text{-}Min_{i,j}/Max_{i,j}\text{-}Min_{i,j} \quad (15)$$

Where:

i = V irtualclient (1, 2, ..., m)

j = Resources (1, 2, 3)

Step 2: Initially the weights are assigned as 1, $W_{i,j,s}(0) = 1$ with category set as 1 (s = 1).

Step 3: Determine the dynamics of Fuzzy ART network, where $\alpha\epsilon[0, 1]$: ChoiceParameter it controls the choice of category whose weight vector W_{ij} is the largest coded subset of I.

$\beta\epsilon[0, 1]$: LearningRate, it defines the degree to which weight vector W_{ij} is updated. $\rho\epsilon[0, 1]$: V igilanceParameter, it defines the required level of similarity of patterns within clusters.

Step 4: To categorize the input virtual clients with the specified usage of resources, the output node receive net input in the form of choice function for individual client as T_i, defined in Theorem 4.3 for any client in the cloud the choice function value:

$$T_i = |(NI_{i,j} \wedge W_{ij})|/(\alpha+|W_{ij}|); \forall j \in \{1, 2, 3) \quad (16)$$

Where $(a \wedge b)_i = Min(a_i, b_i)$; $|A| \equiv \sum^m |a_i|$.

Step 5: The winning category of individual client i whose maximal T_i is found based on the resource usage limitation, $T_i = max(T_i: i = 1, 2, ..., m)$. If T_i is maximum for more than one category iwith smallest index is chosen.

Step 6: To accept the virtual client it should be nominated to a particular category the matching function, it should exceed the vigilance parameter (ρ):

$$M_{i,s} = |(NI_{i,j} \wedge W_{ij})|/|NI_{i,j}| \quad (17)$$

If $M_{(i,s)} = \rho$, then pass that virtual client to existing category. Else if $M_{(i,s)} < \rho$, then create a new category C_{s+1}.

Step 7: The weight factor winning category is updated as follow:

$$W_j^{new} = \beta(NI_{i,j} \wedge W_j^{old}) + (1-\beta)(NI_{i,j} \wedge W_j^{old}) \quad (18)$$

Step 8: This algorithm is repeated for each virtual client from step 4-7, finally the virtual clients are categorized into 4 categories as Secure, Vulnerable, Modified, Anomaly.

Prediction of user behaviour trust using Neuro fuzzy system: The neuro fuzzy technique is a knowledge based method that uses the fuzzy systems with learning artificial networks. The learning procedure operates on local information and it causes only local modifications in underlying fuzzy systems. It is represented with set of fuzzy rules at any time of learning process. The categories obtained in Fuzzy ART are now trained with neuro fuzzy system (Zadeh, 1973). From those two categories, only two are taken into consideration such as modified and vulnerable. From the above observation, it is known that the virtual clients that are under secure are trusted clients. Inversely the anomalies are under the control of attackers. Since, the trust rate evaluation is unnecessary. So, the remaining vulnerable and modified type categories are interacted with learning environment. Trust rate of all virtual clients are obtained using neuro fuzzy algorithm. The following are the steps to be followed for neuro fuzzy implementation.

Fuzzy ART algorithm:

Require: i = Clients, j = resources
for each I(i, j)
Normalize NI(i, j)
for all I(i, j) with s = 1
set $W_{i,j,s}(0) = 1$
for each NI(i ,j)$\in$[0, 1]
Add vector to the network
compute $T_{i,s}$
Find Max$T_{i,s}$
Compute $M_{i,s}$
if $M_{i,s}>\rho$, then
Add to existing category C_s
else if $M_{i,s}<\rho$
Reset $T_{i,s}$ = -1, Go to step 9
Create new category C_{s+1}
end if
Update weight $W_{i,j}^{new}$
Goto step 5 for next input

Layer 1: It is a process of identifying input/output variables such as Disk space, Memory and GFLOPS. These inputs are used to assign a meaningful linguistic states and their ranges. To prefer exact linguistic states for each variable and pose them by corresponding fuzzy sets. These linguistic states are proposed as fuzzy sets (or) fuzzy numbers. Which consider that the ranges of input variables Disk space, Memory and GFLOPS are [0, 1] and respectively and the range of output variable are trust rate is [0, 1]. The linguistic input variables are Disc Space, Memory, GFLOPS and output variable is trust rate. Each node in this layer which corresponds to one input variable, only transmits input values to next layer.

Layer 2: In order to construct the membership function, each input is simulated with the fuzzy set to provide convenient method for resource usage. It is used for evaluating general virtual clients states in cloud. This layer processes each input for virtual client with their membership function. Each node in this layer corresponds to one linguistic label represented in table to each of the input variables in layer 1. The output link represents the membership value which specifies the degree to which an input value belongs to a fuzzy set is calculated in layer 2. Consider a fuzzification function of the form:

$$f_d: [-a, a] \rightarrow R \quad (19)$$

where, R denotes the set of all fuzzy numbers and $f_d(x_0)$ is a fuzzy number chosen by f_d as approximation of the measurement $d = x_0$. We introduced trapezoidal shape as membership function to define $f_d(x_0)$. It is showing the two control variables and their trapezoidal view to represent fuzzy numbers.

Layer 3: A node in this layer represents the antecedent part of a rule. Usually T-Norm is a operator is used in this node. In our approach Disk space, Memory and GFLOPS are inputs form a set of numbers that are organized with their linguistic states. Trust rate is output variable then:

$$\text{If } d = A \text{ and } \bar{d} = B, \text{ then } \tau = C \quad (20)$$

where, A-C are fuzzy numbers chosen from the set of numbers and their linguistic states. The possible rule generated for each input and output variable is 3. To find the fuzzy rules practically we need a set of input-output data of the following:

$$X\{<x_k, y_k, z_k>|k\in K\} \quad (21)$$

Where:

z_k = A attained value of output variable τ for given value x_k and y_k of the input variable d and $\bar{d}$, respectively

K = An appropriate index set

Let $A(x_k)$, $B(y_k)$, $C(z_k)$ denote the largest membership grades. Then the degree of relevance can be expressed by:

$$i_1[i_2(A(x_k), B(y_k), C(z_k)] \quad (22)$$

where, i_1, i_2 are t-norms. A function i: $[0, 1]^2 \rightarrow [0, 1]$ such that for all a, b, d$\in$[0, 1]; i(a, 1) = a; b$\leq$d implies i(a, b)$\leq$(a, d); i(a, b) = i(b, a); i(a, i (b, d)) = i(i(a, b), d).

The function is usually also continuous and such that i(a, a)$\leq$a for all a$\in$[0, 1]. The output of the layer 3 represents the string length of the corresponding rule.

Memory vs. GFLOPS:

- IF GFLOPS is "low" AND Memory is "low" THEN Trust rate is "Deep"
- IF GFLOPS is "low" AND Memory is "medium" THEN Trust rate is "Deep"
- IF GFLOPS is "low" AND Memory is "high" THEN Trust rate is "Shallow"
- IF GFLOPS is "medium" AND Memory is "low" THEN Trust rate is "Deep"
- IF GFLOPS is "medium" AND Memory is "medium" THEN Trust rate is "Average"
- IF GFLOPS is "medium" AND Memory is "high" THEN Trust rate is "Shallow" IF GFLOPS is "high" AND Memory is "low" THEN Trust rate is "Average"
- IF GFLOPS is "high" AND Memory is "medium" THEN Trust rate is "Shallow"
- IF GFLOPS is "high" AND Memory is "high" THEN Trust rate is "Shallow"

Disk space and GFLOPS:

- IF Disk Space is "very low" AND GFLOPS is "low" THEN Trust rate is "Deep"
- IF Disk Space is "very low" AND GFLOPS is "medium" THEN Trust rate is"Deep"
- IF Disk Space is "very low" AND GFLOPS is "high" THEN Trust rate is "Shallow" IF Disk Space is "low" AND GFLOPS is "low" THEN Trust rate is "Deep"
- IF Disk Space is "low" AND GFLOPS is "medium" THEN Trust rate is "Average"
- IF Disk Space is "low" AND GFLOPS is "high" THEN Trust rate is "Shallow"
- IF Disk Space is "medium" AND GFLOPS is "low" THEN Trust rate is "Average"
- IF Disk Space is "medium" AND GFLOPS is "medium" THEN Trust rate is "Average"
- IF Disk Space is "medium" AND GFLOPS is "high" THEN Trust rate is "Average"
- IF Disk Space is "high" AND GFLOPS is "low" THEN Trust rate is "Average"
- IF Disk Space is "high" AND GFLOPS is "medium" THEN Trust rate is "Shallow"
- IF Disk Space is "high" AND GFLOPS is "high" THEN Trust rate is "Average"
- IF Disk Space is "very high" AND GFLOPS is "low" THEN Trust rate is "Shallow"
- IF Disk Space is "very high" AND GFLOPS is "medium" THEN Trust rate is "Shallow"
- IF Disk Space is "very high" AND GFLOPS is "high" THEN Trust rate is "Shallow"

Disk space and Memory:

- IF Disk Space is "very low" AND Memory is "low" THEN Trust rate is "Deep"
- IF Disk Space is "very low" AND Memory is "medium" THEN Trust rate is "Deep"
- IF Disk Space is "very low" AND Memory is "high" THEN Trust rate is "Average"
- IF Disk Space is "low" AND Memory is "low" THEN Trust rate is "Deep"
- IF Disk Space is "low" AND Memory is "medium" THEN Trust rate is "Average"
- IF Disk Space is "low" AND Memory is "high" THEN Trust rate is "Average"
- IF Disk Space is "medium" AND Memory is "low" THEN Trust rate is "Deep"
- IF Disk Space is "medium" AND Memory is "medium" THEN Trust rate is "Average"
- IF Disk Space is "medium" AND Memory is "high" THEN Trust rate is "Shallow"
- IF Disk Space is "high" AND Memory is "low" THEN Trust rate is "Average"
- IF Disk Space is "high" AND Memory is "medium" THEN Trust rate is "Shallow"
- IF Disk Space is "high" AND Memory is "high" THEN Trust rate is "Shallow"
- IF Disk Space is "very high" AND Memory is "low" THEN Trust rate is "Average"
- IF Disk Space is "very high" AND Memory is "medium" THEN Trust rate is "Shallow" IF Disk Space is "very high" AND Memory is "high" THEN Trust rate is "Shallow"

Layer 4: The observation of input variable to find the correctness of inference rules, is matched with fuzzy inference rules to make inference in terms of output variables (Zadeh, 1973). We choose composite inference logic mentioned in Definition 4 to define our variables. We convert the given fuzzy inference rules represented which is equivalent to simple fuzzy conditional proposition of the form:

$$\text{If} \langle d, \bar{d} \rangle \text{ is } A \times B, \text{ then } \tau \text{ is } C \quad (23)$$

Where:

$$[A \times B](x, y) = \min[A(x), B(y)] \quad (24)$$

for all x∈[-a, a] and y∈[-b, b]. The output variable trust rate τ becomes the problem of approximate reasoning with composite inference fuzzy proposition mentioned in definition 3 and 4, respectively. The fuzzy rule base consists of 'n' fuzzy inference values, then:

$$\begin{aligned}
&\text{Rule 1: IF}(, \bar{d}) \text{ is } A_1 \times B_1, \text{ then } \tau \text{ is } C_1 \\
&\text{Rule 2: IF}(d, \bar{d}) \text{ is } A_2 \times B_2, \text{ then } \tau \text{ is } C_2 \\
&\ldots \\
&\text{Rule n: IF}(d, \bar{d}) \text{ is } A_n \times B_n, \text{ then } \tau \text{ is } C_n \\
&\text{Fact: } (d, \bar{d}) \text{ is } f_d(x_0) \times f\bar{d}_{(y0)} \\
&\ldots \\
&: \tau \text{ is } C
\end{aligned}$$

The classes deep, average, shallow denote fuzzy sets that represent the linguistic states of input variables disk space, memory, GFLOPS, respectively.

The rule is explained in terms of relation R_j, which is mentioned in definition 2. The rules are considered as disjunctive in nature. We derive the Eq. 16 to conclude the output variable τ is defined by the fuzzy set as:

$$C = [f_d(x_0 \times f_{\bar{d}}(y_0)] o^i R_j \quad (25)$$

where, o^i is the sup-icomposition for a t-norm i. The choice of the t-norm is a matter similar to the choice of fuzzy sets for given linguistic labels.

Layer 5: The process of computing single fuzzy number from C is called Defuzzification. The fuzzy output variable is also a linguistic variables whose value is calculated by assigning grades of membership. In the previous step, we found a single number compatible with membership function in trust rate and it is called output membership function. This number will be the output for the final step in defuzzification process. There are several methods for calculating a single defuzzified number. We use a centroid method to convert the output values of inference engine as crisp numbers and they are expressed as fuzzy set. We calculate the output variable by using centroid method and it is expressed as:

$$x^* = \frac{\int_a^b \mu_A(x)xdx}{\int_a^b \mu_A(x)dx} \quad (26)$$

Let $\mu_A(x)$ be the corresponding grade of membership in the aggregated membership function, let:

- X_{min} be the minimum x value attain the minimum of trust rate τ
- X_{mod} be the moderate x value attain the moderate of trust rate τ
- X_{max} be the maximum x value attain the maximum of trust rate τ
- X^* is defuzzified output as a real number value

RESULTS AND DISCUSSION

Fuzzy ART analysis: We analyzed the proposed method to detect the abnormal handling of resources by the virtual clients in the cloud using the technique Fuzzy ART. This experiment is conducted in the MATLAB 7.1 and the results are interpreted. The Fuzzy ART is modelled with input factors such as Memory, Disk space and GFLOPS. The fuzzy ART configuration uses three predetermined values such as vigilance threshold, choice parameter and learning rate. In Fuzzy ART the lower the vigilance value struggle to precisely separate input patterns belonging to different input classes whereas higher the vigilance values favors good but result in multiple categories representing single input classes. The data interpreted in Table 1 are trained by Fuzzy ART with above predefined parameters. These are sample data in a single clouds provider but in a cloud there are millions of clients are available and they are clearly categorized with this learning technique. To evaluate our system, sample statistics of virtual clients are categorized based on the limit in their usage of resources allocated to them. The percentage of such categories in the network has a direct

Table 1: Input parameter for Fuzzy ART training

Resources	V C1	V C2	V C3	V C4	V C5
Memory	0.1818	1	0.0013	0.2727	0.1515
Disk space	0.3784	0.9981	0.3538	0.0031	0.041
GFLOPS	0.0012	0.9998	0.1085	0.0182	0.0012

Table 2: Categories of Fuzzy ART

Virtual clients	Categories
VC2	Secure
VC3, VC5	Vulnerable
VC2	Modified
VC4	Attacker

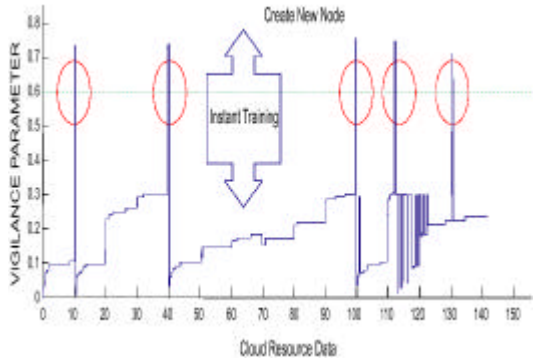

Fig. 3: Fuzzy ART-vigilance training

relationship with the network vigilance and performance. The greater accuracy in categorizing is evaluated using Matching test function to finely categorize. The output is categorized into 4 classes [Secure, Vulnerable, Modified, Anomaly] (Fig. 3).

The results in Table 2 clearly illustrate Fuzzy ART outperforms better than other learning techniques under tested conditions. Based on these results it is clear that the Fuzzy ART configuration were able to achieve a training accuracy of hundred percentage with only 4 categories. This allows it to learn a new input with minimum impact on its existing knowledge and accuracy. Thus categorization used in Fuzzy ART is far better than other techniques with higher accuracy. The categories thus classified are then learned with Neuro Fuzzy system. Since the virtual clients under secure are safe enough in cloud and the clients in anomaly are hacked by the attackers, they are not analyzed with Neuro Fuzzy. The categories such as vulnerable and modified are learnt with neuro fuzzy system and trust rate for every client is analyzed.

Neuro fuzzy analysis: The virtual clients in the vulnerable and modified categories are learnt with neuro fuzzy system and trust rate is analyzed. In this technique Mamdani FIS is adapted as it is more effective for system identification. The trust rate is calculated based on the consumption of resources by the system. For estimation

of the trust rate, membership functions of the input variables are selected to satisfy the desired output. A multilayer feed forward neural network trained is used to adjust the membership function parameters according to the input-output characteristic of training patterns. The trust rate depends on the number of rules which on other hand depend on the number of membership functions and inputs. This model performs well when the resource usage is low for the training set. It can be seen that the model uses only three input variables (Memory, GFLOPS, Disk space). The chosen input has membership functions that are presented over a linguistic universe such as low,medium, high for memory and GFLOPS. Similarly for Disk space it is represented with four-valued linguistic universe such as very low, low, moderate, high and very high. As a result the trust rate for comparison of memory and GFLOPS uses 9 rules, memory with Disk space and GFLOPS uses 15 rules. The fuzzy rules are a valuable source of information from which knowledge can be extracted that governs the relationship between virtual client and the factors affecting clients. A review of the data used indicates that almost of fifty percentage of data were grouped as low trust rate in both group but one or two virtual clients have high trust rate. So, the systems with high trust rate can be prevented from attack by attackers and they can be survived under secure group from vulnerable. Similarly the system with low trust rate can be monitored to prevent it from the control of attacker. The above results indicate the neurofuzzy networks have the ability to extract rules from the data that make physical sense (Fig. 4 and 5).

A comprehensive comparison of fuzzy control and neuro fuzzy ART is done. The neuro fuzzy ART performs fast stable learning and categorization of analog patterns by an adaptive resonance system for different behaviors (Fig. 6). It demonstrates that it performs the classification without any interception of predefined data sets, i.e., it follows unsupervised data training. In case of fuzzy control it performs supervised learning, so the efficiency of neuro fuzzy ART performs better. Every virtual client is trained by neuro fuzzy ART and its trust rate evaluation is done based on the resource consumption. This system shows that the trust model suggested is more accurate in predicting the trust in virtual cloud environment rather than fuzzy control. The configurations that were discovered in Fuzzy ART shows it efficient classification of inputs correctly. The virtual clients under certain categories are tested with neuro fuzzy system to evaluate the trust rate. Based on these results, it is clear that the tested neuro fuzzy ART configurations were able to achieve higher performance than fuzzy control system which achieved the more

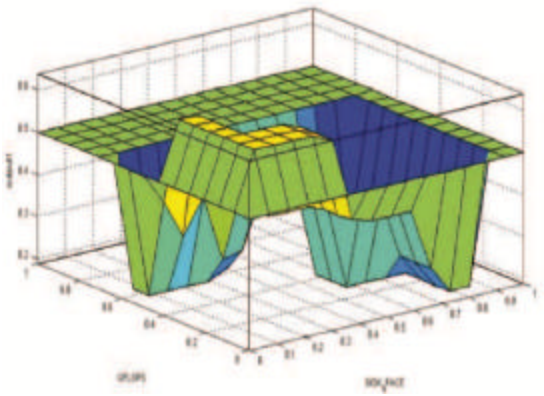

Fig. 4: Neuro fuzzy view of GFLOPS and disc space vs. trust rate

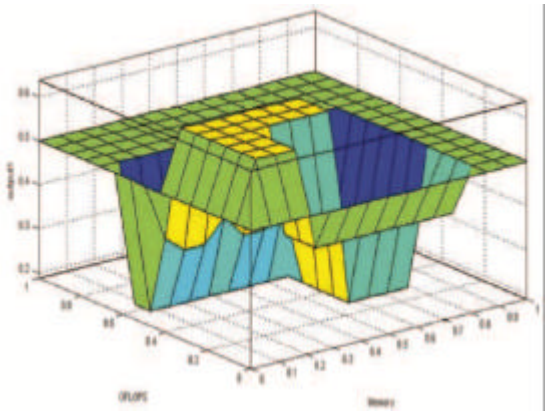

Fig. 5: Neuro fuzzy view of GFLOPS and memory vs. trust rate

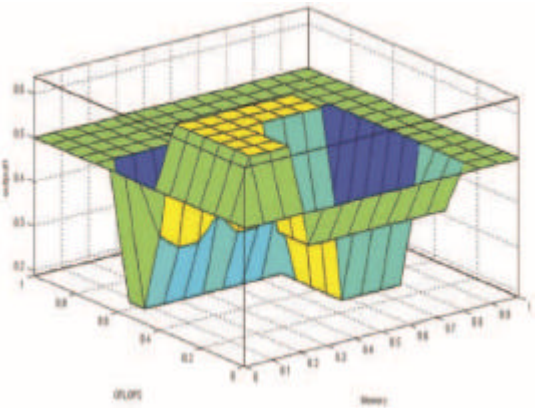

Fig. 6: Neuro fuzzy view of memory and disc space vs. trust rate

training accuracy. We record the quality of our system and made a comparison of our scheme with

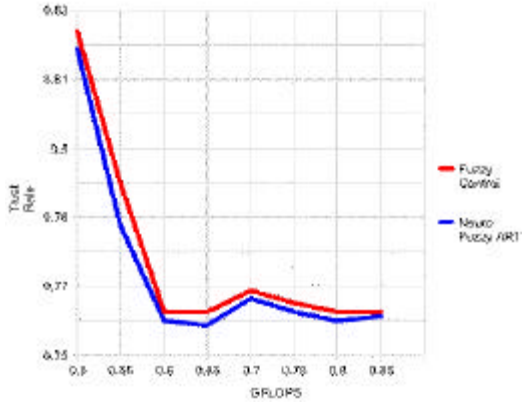

Fig. 7: Neuro Fuzzy ART vs. Fuzzy Control-GFLOPS Trust rate comparison

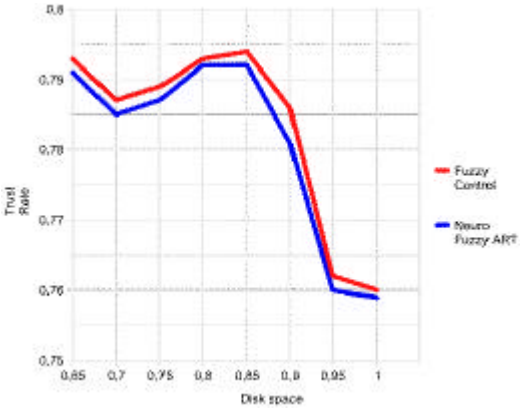

Fig. 8: Neuro fuzzy art vs. Fuzzy control-disc space trust rate comparison

(H2FTM) Hypervisor Hardware Fuzzy Trust Monitor (Jaiganesh and Kumar, 2013). It was examined that they used fuzzy control to predict the trust rate of the virtual client. In this experiment, they considered the GFLOPS and Disc space as input factor to fuzzy control instances. It also shows that it trained the complex data sets than the fuzzy control system. The results of two systems are plotted in Fig. 7 and 8. It was analyzed that the trust evaluated is more accurate in neuro fuzzy ART for consumption of resources such as GFLOPS and disc space than fuzzy control system.

CONCLUSION

There are diversified of attacks that pivot on the issue of hypervisor and cloud computing. We presented a resource based trustworthiness of clients in virtual environment called virtual client behavior trust. We have modelled neuro fuzzy ART system to predict the user beahviour trust. This structure can be used by any hypervisors to manage and forecast their self against resource based attacks. Neuro fuzzy ART offers to synchronize the virtual environment as trusted system. The advantage of proposed system is in incrementing the hypervisor introspection functionalities and being free from malware in virtual backgrounds. It provides dynamic data protection to identify the trustworthiness of clients and character reliability. It offers back up awareness to provide an alarm for restoring and blocking hypervisor services. The research will be extended to provide secure transactions against the services offered by hypervisor via cloud computing.

REFERENCES

Armbrust, M., A. Fox, R. Griffith, A.D. Joseph and R.H. Katz *et al*., 2009. Above the clouds: A Berkeley view of cloud computing. Technical Report No. UCB/EECS-2009-28, Department of Electrical Engineering and Computer Science, University of California, Berkeley, February 10, 2009. http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html.

Bendale, Y. and S. Shah, 2013. User level trust evaluation in cloud computing. Intl. J. Comput. Appl., 69: 31-35.

Blume, M. and S.C. Esener, 1997. An efficient mapping of fuzzy ART onto a neural architecture. Neural Netw., 10: 409-411.

Buyya, R., C.S. Yeo and S. Venugopal, 2008. Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities. Proceedings of the HPCC'08. 10th IEEE International Conference on High Performance Computing and Communications, September 25-27, 2008, IEEE, Dalian, China, ISBN: 978-0-7695-3352-0, pp: 5-13.

Carpenter, G.A., S. Grossberg and D.B. Rosen, 1991. Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. Neural Networks, 4: 759-771.

Dutreilh, X., N. Rivierre, A. Moreau, J. Malenfant and I. Truck, 2010. From data center resource allocation to control theory and back. Proceedings of the 2010 IEEE 3rd International Conference on Cloud Computing (CLOUD), July 5-10, 2010, IEEE, Miami, Florida, ISBN: 978-1-4244-8207-8, pp: 410-417.

Frank, T., K.F. Kraiss and T. Kuhlen, 1998. Comparative analysis of Fuzzy ART and ART-2A network clustering performance. IEEE Trans. Neural Networks, 9: 544-559.

Habib, S.M., S. Ries, M. Muhlhauser and P. Varikkattu, 2014. Towards a trust management system for cloud computing marketplaces: Using caiq as a trust information source. Secur. Commun. Netw., 7: 2185-2200.

Huang, J. and D.M. Nicol, 2013. Trust mechanisms for cloud computing. J. Cloud Comput., 2: 1-14.

Huang, J., M. Georgiopoulos and G.L. Heileman, 1995. Fuzzy ART properties. Neural Netw., 8: 203-213.

Ibrahim, L.M., 2010. Anomaly network intrusion detection system based on Distributed Time-Delay Neural Network (DTDNN). J. Eng. Sci. Technol., 5: 457-471.

Iosup, A., S. Ostermann, M.N. Yigitbasi, R. Prodan, T. Fahringer and D.H. Epema, 2011. Performance analysis of cloud computing services for many-tasks scientific computing. IEEE Trans. Parallel Distrib. Syst., 22: 931-945.

Jaiganesh, M. and A.V.A. Kumar, 2012. JNLP Based Secure Software as a Service Implementation in Cloud Computing. In: Mathematical Modelling and Scientific Computation. Balasubramaniam, P. and R. Uthayakumar (Eds.). Springer Berlin Heidelberg, Berlin, Germany, ISBN: 978-3-642-28925-5, pp: 495-504.

Jaiganesh, M. and A.V.A. Kumar, 2013. B3: fuzzy-based data center load optimization in cloud computing. Math. Prob. Eng., Vol. 11,

Jaiganesh, M. and R. Sivasankari, 2012. ACDP: Prediction of application cloud data center proficiency using fuzzy modeling. Proc. Eng., 38: 3005-3018.

Keskin, G.A., S. Ýlhan and C. Ozkan, 2010. The Fuzzy ART algorithm: A categorization method for supplier evaluation and selection. Expert Syst. Appli., 37: 1235-1240.

Kuehnhausen, M., V.S. Frost and G.J. Minden, 2012. Framework for assessing the trustworthiness of cloud resources. Proceedings of the 2012 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), March 6-8, 2012, IEEE, New Orleans, Louisiana, USA., ISBN: 978-1-4673-0343-9, pp: 142-145.

Lin, C.J. and C.T. Lin, 1997. An ART-based fuzzy adaptive learning control network. Fuzzy Syst. IEEE. Trans., 5: 477-496.

Lombardi, F. and R. Di Pietrob, 2011. Secure virtualization for cloud computing. J. Network Comput. Appl., 34: 1113-1122.

Luo, M., T. Peng and C. Leckie, 2008. CPU-based DoS attacks against SIP servers. Proceedings of the Symposium on Network Operations and Management Symposium (NOMS) 2008, April 7-11, 2008, IEEE, Salvador, Brazil, ISBN: 978-1-4244-2065-0, pp: 41-48.

Modi, C., D. Patel, B. Borisaniya, H. Patel and A. Patel *et al*., 2013. A survey of intrusion detection techniques in cloud. J. Netw. Comput. Appl., 36: 42-57.

Ngamwitthayanon, N. and N. Wattanapongsakorn, 2011. Fuzzy-ART in network anomaly detection with feature-reduction dataset. Proceedings of the 2011 The 7th International Conference on Networked Computing (INC), September 26-28, 2011, IEEE, Gyeongsangbuk-do, South Korean Province, ISBN: 978-1-4577-1129-9, pp: 116-121.

Patel, A., M. Taghavi, K. Bakhtiyari and J.C. Junior, 2013. An intrusion detection and prevention system in cloud computing: A systematic review. J. Netw. Comput. Appl., 36: 25-41.

Rong, C., S.T. Nguyen and M.G. Jaatun, 2013. Beyond lightning: A survey on security challenges in cloud computing. Comput. Electr. Eng., 39: 47-54.

Scarfone, K.A., 2011. Guide to Security for Full Virtualization Technologies. DIANE Publishing, Collingdale, Pennsylvania, USA., Pages: 125.

Shea, R. and J. Liu, 2012. Understanding the impact of denial of service attacks on virtual machines. Proceedings of the 2012 IEEE 20th International Workshop on Quality of Service, June 4-5, 2012, IEEE Press, Piscataway, New Jersey, USA., ISBN: 978-1-4673-1298-1, pp: 1-27.

Tian, L.Q., C. Lin and Y. Ni, 2010. Evaluation of user behavior trust in cloud computing. Proceedings of the 2010 International Conference on Computer Application and System Modeling (ICCASM), October 22-24, 2010, IEEE, Taiyuan, China, ISBN: 978-1-4244-7235-2, pp: V7567-V7572.

Wang, C., K. Ren and J. Wang, 2011. Secure and practical outsourcing of linear programming in cloud computing. Proceedings of the 2011 IEEE Conference on INFOCOM, April 10-15, 2011, IEEE, Shanghai, China, ISBN: 978-1-4244-9919-9, pp: 820-828.

Wang, C., Q. Wang, K. Ren and W. Lou, 2010. Privacy-preserving public auditing for data storage security in cloud computing. Proceedings of the 2010 IEEE Conference on INFOCOM, March 14-19, 2010, IEEE, San Diego, California, pp: 1-9.

Worku, S.G., C. Xu, J. Zhao and X. He, 2014. Secure and efficient privacy-preserving public auditing scheme for cloud storage. Comput. Electr. Eng., 40: 1703-1713.

Xue, J., M. Li, W. Zhao and S.Y. Chen, 2012. Bound maxima as a traffic feature under DDOS flood attacks. Math. Prob. Eng. Vol. 20,

Zadeh, L.A., 1973. Outline of a new approach to the analysis of complex systems and decision processes. IEEE Trans. Syst. Man Cybernet., 3: 28-44.