ACTIVE BAYES CLASSIFIER MODEL FOR MULTI-TENANT TRUST MAINTENANCE ON SOFTWARE SERVICE PROVIDENCE

K. Chidambaram[1] *C. Chandrasekar*[2]

ABSTRACT

Cloud based trust is regarded as the essential secure relationship system on software application to applicationinteraction. Current works on the cloud software trust model does not reasonably and precisely reflect some important characteristics such as authenticating the client side requests.A key issue namely the usage unauthorized service executables are not controlled in the cloud system trust management.Using a few simple methods, the user services are authenticated, however the security issues arise on the multiple tenants.Software Trust development methodology is a hot topic in the cloud environment to solve the security issues and enable the high protection on the software services. To improve the software trust level on the cloud Infrastructure, Multi-Tenant Active Bayes Classifier Trust Maintenance(MTM-ABC) model is proposed in this paper.MTM-ABC model works to self-manage the trust level on dual approach (i.e., client and server side) while providing the software services.MTM-ABC model works under the two phases namely MTM-ABC- Client model and MTM-ABC-Cloud Server model. Active Bayes Classifier in the client model dynamically makes the trust decision in the different situation of the user request to the software services. The active client trust evaluation function is used in the MTM-ABC- Client model to compute the authentication level based on services provider through prior knowledge. On the MTM-ABC-Cloud Server model, the authentication to the multiple tenants is provided simultaneouslyusing Eigen Server Trust Function. The Eigen Server Trust function computed on all the tenants based on the history information. The unauthenticated request from the client side is discarded, thereby decreasing the downloadmemoryof the unauthenticated software services. Experiment is conducted on factors such as average response time, trust level maintenance rate on simultaneousmulti-tenants request and throughput level on accessing the software from cloud zone.

Keywords: Multi-Tenant, Cloud infrastructure, Software Trust Level, Active Bayes Classifier, Eigen Server Trust Function, Active Client Trust Evaluation

I. INTRODUCTION

In particular, cloud computing involve the practice of an ordinary IT infrastructure and services provided to the diverse tenants. This implies there is a need of a strong

[1]Research Scholar,Computer Science Department, Karpagam University, Coimbatore, 642120, India. Email:chidamkl@gmail.com Mobile: +91-9841783807

[2]Associate Professor, Computer Science Department, Periyar University, Salem, 636011, India. Email: ccsekar@gmail.com Mobile: +91-9994599967

design of security boundaries in order to separate the unauthorized tenants. Thus, the system controls the access to the unauthorized tenants in order to providean efficient control over the usage of the cloud architecture. The accessibility of advanced authorization capacity is a differentiating feature for cloud providers, which demand the design of suitable authorization models (trust model on client and server machines) to enhance the access control on the cloud zone.

Cloud scheduler in [1], focus on providing the physical and virtual resources to offer the user requirements from the server machine.OpenStack based prototype identifies the most complex component request, however the trustworthy collection of requests are not collected with the properties. The clouds' trustworthy self-managed software services are not established.Object-centered approach [2]was designed with the logging mechanism to ensure the accountability for data sharing the cloud.This approach aimed at providing software tamper conflict to facilitate the protection but security policies are not employed.

Trust enhanced security model, [3] for cloud servicing to multiple clients assist to detect and avoid security attacks in cloud infrastructures using trusted attestation techniques. Trust enhanced security model facilitate the cloud service provider to confirm assured security properties of the tenant virtual machines and services. Remote Data Auditing Technique [4] was designed for auditing the system on the security parameter on single server. This method provides the optimal and lightweight security framework but the dynamic updation of the data and data access control is not presented.Mandatory Access Control (MAC) and Role Based Access Control (RBAC) was developed in [5] for addressing the access control model in cloud environment.Access control model meet the identified cloud access control requirements and ensure the secure sharing of information betweenthe client and server machine.

The access control model, [6] was developed for multi-tenancy authorization as well as the rules employed to protect the accessin the cloud. Access control system is potentially suitable for all layers of the cloud computing stack but not extended the system by integrating with the Openstack.Fine-grained trust model manages the federations in cloud computing environments but moreover the advanced trust model is also not employed for the suitable cloud computing operations.Trust Assessment Module (TAM) on Cloud Service Provider (CSP), in [7], provide preventive control by using the detective mechanism. TAM intended to control mutual policy management for the resource utilization and remove the computational resource barriersfor high trustworthy providence.

In this work, focus is made on Multi-Tenant Active Bayes Classifier Trust Maintenance (MTM-ABC) model to improve the software trust on multi tenants servicing. Contribution of the work is as follows,

- To self-manage the trust level, dual approach is developed on the cloud infrastructure.
- To dynamically makes the trust decision in the different situation of the tenant request, Active Bayes Classifier is employed
- To compute the authentication level on client side, active client trust evaluation function is used based on services provider through prior knowledge.
- To compute the authentication level on server side, Eigen Server Trust Function is computed based on the history information.
- To secure the servicing from the cloud infrastructure, trust function evaluated values are used.

The structure of this paper is as follows. In Section 1, describes the drawback of the cloud software trust model on servicing to the multiple tenants. Section 2 demonstrates the related cloud servicing to the tenants. In Section 3, an overall description of Multi-Tenant Active Bayes Classifier Trust Maintenance (MTM-ABC) model is presented.Section 4 and 5 sketch experiment results with parametric factors and present the result graph for research on software services on the cloud infrastructure.Finally, Section 6 concludes the work.

II. Related Works

As the growth of the computing infrastructure in the information technology field, the security issues occur in the larger wide.The security issues and vulnerabilities in the cloud environment were briefly explained in [8] and recognize the unprotected threats.The security model of cloud provider boundary isevaluated. But the standard secure cloud environmentis not available in the present techniques.Security issues in cloud are briefly explained with the security models [9], and discovered a robust system for the effective transaction. But issues occur on ensuring the effective authentication based scheme in the cloud environment. Virtualization technique [10] was described to provide guarantees on the protection of privacy sensitive data stored in the cloud server. The virtualization is introduced to attain the security on sharing the data to the multiple tenants. However, it failson developing the Homomorphic Encryption cryptosystem for maintain the security on the different size of the message blocks.

To control the memory management by the cloud users, Group-based customization of memory with KVM-based prototype in [11] is addressed. Group-based memory deduplication and reprovisioningmake sure the firm inter-group isolations. The cloud wide grouping based on the memory fails on enhances the memory efficiency by reducing underutilized surplus memory. Merkle Hash Tree was designed in [12] tosustaincompetent integrity inspection and file information update in the cloud infrastructure.Hierarchical key organization attainsthe suitable keys management and proficient permission revocation. Shield maintainssimultaneous write access by employing aneffective linked list but the Proxy server group is not addressed in this method.

Digital forensics in cloud computing [13], addresses as an effective digital storage of the criminal activities. However the security problem arises on transferring the solution to the criminal activities from the cloud server to the client systems. However the progression of the connectivity between the cloud environment systems is not addressed. Trusted Computing Platform (TPM) [14]

was designed prevents the forgery of fake key events in cloud systems which attacked by malware. But the construction of the application level data verification is not addressed in TPM. Trusted Third Party [15], tasked with assureprecise securitycharacteristics calls the Public Key Infrastructure operations within a cloud environment. This trust worthy environment make sure the authentication, integrity and confidentiality of involved data and communications. However the focus towards the quality of services is not addressed.

III. Multi-Tenant Active Bayes Classifier Trust Maintenance On Cloud Infrastructure

In a cloud computing environment, Multi-Tenant Active Bayes Classifier Trust Maintenance (MTM-ABC) model is developed to improve the trust level on sharing the software stacks. Trust is regarded as an essential secured relationship within the cloud environment on providing the services to the client systems from the server zone. The proposed research work is to accurately develop an accurate algorithm for reflecting the essential characteristics of the trust with a dual approach. The dual approach handles the client and server side trust maintenance on sharing the software to the multiple tenants.

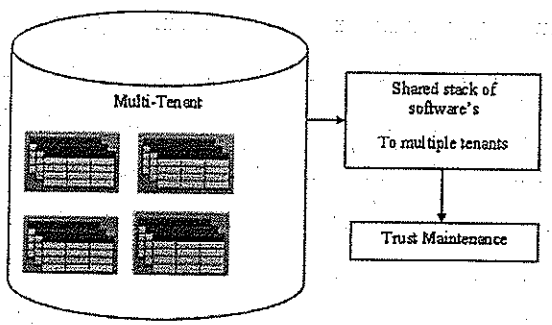


Figure 1 : Multi-Tenant Trust Maintenance

Trust in cloud infrastructure measure the eagerness to believe in an entity based on its competence and behavior at a specified time.Multi-tenant trust maintenance as described in proposed Fig 1 distributes the particular instance of the software application with high security factor. The software shared to the every tenant (i.e., clients) requests. A tenant in MTM-ABC model is able to customize some part of the business rule application as per the client needs.Architecture Diagram of Multi-Tenant Active Bayes Classifier Trust Maintenance (MTM-ABC) model is described in Fig 2.

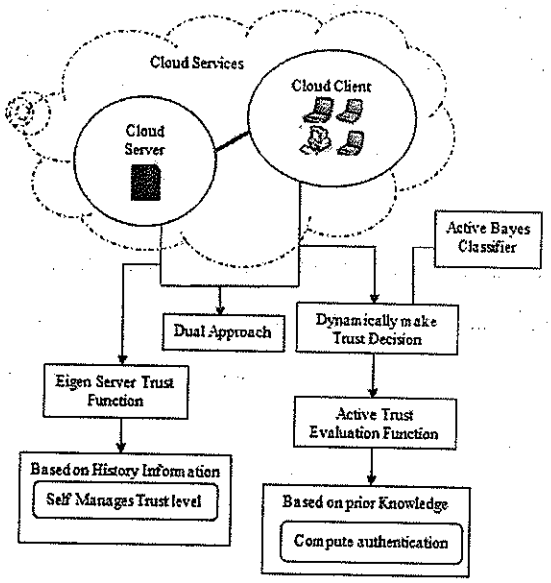


Figure 2 : Diagram of MTM-ABC Model

The interaction among the entities depends on the relationship of trust between each other on the cloud environment to improve the security level. The security level is improved in our proposed work by developing the MTM-ABC Model. The proposed model takes the cloud services with client and server machines. The client request to the software stacks and the client is

authenticated by the server using the proposed trust model. The server machine shares the software to the requested multiple tenants with the high security level. The dual approach is advised in MTM-ABC Model to handle the client and server trust functions.

The server measures the trust level using the Eigen Server Trust Function. Eigen Server Trust Function is employed to measure the authenticity of the multiple tenants on the cloud infrastructure. Requests from the different variety of tenants need to compute the trust function in MTM-ABC Model. Eigen Server Trust Function uses the history function for self-managing the trust level. Simultaneously, the client machine dynamically makes the trust decision using the Active Bayes Classifier. The main advantage of the Active Bayes Classifier is that it avoids the predefined threshold rate and performs the dynamic updation of the decision to improve the trust level. Active Trust Evaluation Function is computed based on the prior knowledge about the software information.

3.1 Trust Maintenance on Cloud Server

The evaluation of trust on the server side in MTM-ABC Model uses the authority factor.The authority factor value under the multiple tenant software requestsshould achieve superior percentage, so that the high level of security is maintained. The multiple tenant security is associated on the server and client side (i.e., on dual approach). The dual approach in MTM-ABC Model is introduced to maintain the security level on both end of the communication in the cloud infrastructure. The trust level measure on the server side is performed through the Eigen server Trust Function, and described briefly in section 1.1.1.

3.1.1 Eigen Server Trust Function

Eigen Server Trust on cloud infrastructure leads to the probabilistic analysis of the request from the multiple tenants.The trust function applied on the aggregation of the multiple requests to produce the reasonable security result. Eigen Server Trust function applied on the single tenant request is formularized as, *Single Tenant TF = max* (F_{ij} *n)* Where n=0 to 1

..... Eqn (1)

Fitting trust F_{ij} contains the 'n' trust value, where n=0 to 1.Trust Function 'TF' takes the higher security value. The probabilistic analysis of the requests from the multiple clients in MTM-ABC Model is described as,

Multiple Client Request Trust Function = $\sum$

..... Eqn (2)

Eqn (2) clearly aggregates the different tenant's software requests 'r'and identifies the trust function level.The multiple tenant requests are aggregated through the sum function. EigenTrust computes a trust score to identify the malicious requests from the multi-tenants in the cloud infrastructure.The degree of the trust function on the probabilistic analysis also uses the history information to self-manage the trust level on the server side. The Eigen Server Trust Algorithm is illustrated as,

Eigen Server Trust

Begin

Step 1: Request 'r' accepted from the multiple tenants at time't'

Step 2: Tenant request is initially authorized through the Authority Factor

Repeat step for multi tenant's request

Step 3: Trust Function based on Eigen is applied on the single tenant request

Step 3.1: Computed Fitness Trust function *Fij*

Step 3.2: Aggregate the multiple client requests

Step 3.2.1: Summing of the trust measure on server

$$\sum [r_{1,2...n}(TFn)]$$

Step 4:Trust Score identify the malicious tenants

Step 5: Self manage the trust level based on history information.

Until all tenant request are authorized

End

The Eigen server trust is based on the authorized factor in MTM-ABC Model. For every multi-tenant request to the server, computed the Eigen trust function value based on the fitness rate. The fitness response with higher percentage denotes the higher trust score without any malicious. The simultaneous request from the multiple tenants is also handles with the self-management of trust level.

3.2 Trust Maintenance on Client Server

The client retrieves the software stack from the cloud environment after the trust approval. The superior score value of trust achieves the higher security level in MTM-ABC Model. The client machines dynamically create trust assessment based on different situations using the Active Bayes Classifier.

3.2.1 Active Bayes Classifier

In client side, Active Bayes Classifier is developed to dynamically make the trust decision with the independent analysis. The client independently measure the trust score value based on the responded software stack to the tenant machine from the server side in MTM-ABC Model. Active Bayes Classifier step by step process is clearly described in the below diagrammatic structure.

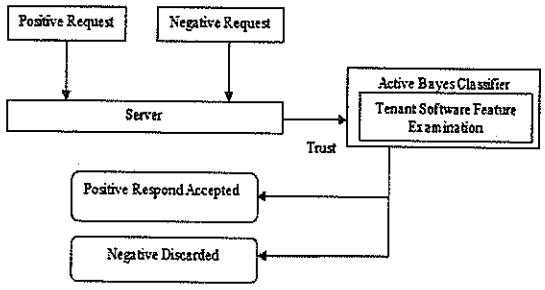


Figure 3 : Processing Step of Trust level using Active Bayes Classifier

The Fig 3 takes the positive and negative request from the multiple tenants' machines. In MTM-ABC Model, server accepts the request by measuring the trust score through Eigen Trust Function. The accepted request is processed to produce the respond result, thereby on the client side the trust measure is carried out. The trust is measured through the active trust evaluation function as briefly explained in section 1.2.2. On the client side, responded software stack from the server measures the trust score.

The tenant receives the software stack and measure the features score to identify whether the server responded accurately to their request through proposed MTM-ABC

Model. Trust score with high value is accepted as the positive respond in our proposed model and all other respond results are discarded on the tenant side. Active Bayes classifier active work in MTM-ABC Model is to dynamically make the trust functions.

3.2.2 Active Trust Evaluation Function

The active trust function is measured based on the prior knowledge of the software stacks between the tenants.The knowledge helps to easily compute the trust score value on the client side. In MTM-ABC Model, the active trust function is formularized as,

$$Services\ (y|x_1, \ldots x_n) \propto Services\ (y)\prod_{i=1}^{n}[P(x_i|y)$$

..... Eqn (3)

The services from the server machine 'y' to the different multiple tenants machinex_1 , x_2 ....x_n evaluate the trust function value. The trust value computed helps deliver the software to the multiple tenants with improved software delivery rate. The maximum delivery rate of software on the client machine inMTM-ABC Model is shown as,

$$y = argmax_{Trust}\ Services(y)\prod_{i=1}^{n}[P(x_i|y)$$

.....Eqn (4)

The services from the server machine in the cloud infrastructure are received and higher trust value 'argmax' produced the multi-tenant machines. The software service sharing rate from the server side is also increased in MTM-ABC Model.

IV. Experimental Evaluation

Multi-Tenant Active Bayes Classifier Trust Maintenance (MTM-ABC) model uses the Cloudsim platform to perform the experimental work.CloudSim is implemented on usingAmazon web service dataset in JAVA platform with a variety of performance factors. Cloudsim goal is to provide a global and extensible simulation framework that facilitates model, simulation, and experimental evaluation. The emerging cloud computing infrastructures and application services allow the users to focus on increasing the software profits on the cloud infrastructure.

The results are investigated with the small stage information which is obtained from experimental work.A data center comprises of many software's with CPU core equivalent to 500, 1000 and 1500 Microprocessor without Interlocked Pipeline Stages (MIPS). The 4GB RAM is used for the experimental work and 1 TB of storage of single domain information. The proposed MTM-ABC model is compared against the existing Innovative Admission Control and Scheduling (IACS) algorithms for SaaS providers and Domain-based Integrity (DR@FT) model.MTM-ABC Mechanism experiments the work on the factors such as average response time, trust level maintenance rate on simultaneous multi-tenants request, overall flexibility of multi-tenant system and throughput level on accessing the software from cloud zone.

V. Result Analysis

To assess the performanceof Multi-Tenant Active Bayes Classifier Trust Maintenance (MTM-ABC) model and compare it with the other systems, namely, Innovative

Admission Control and Scheduling (IACS) algorithms for SaaS providers and Domain-based Integrity (DR@FT) model. All these methods are implemented in the CloudSim simulator. The simulator tool is used to identify the working condition of the proposed system and their performance result percentage before the real world applications.The performance values are investigated which is obtained from the cloud simulator experimental work.

The base scheme used for the trusted data sharing between the client and server machine from the cloud storage area. The machine uses the Java 1.7 on Intel Core-i5 processor with 3 GB RAM. 1 TB of storage of single domain informationis taken to compute the authentication level based on services provider. Used the Virtual Machine (VM) with one server CPU of 4 MB of memory and other 8 VM are considered as the client machines. The single and mixed workload used on authenticating the multiple tenants VM. VM works as a master node and the other VMs work as slave nodes respectively with high efficiency. The configuration is similar on virtual client'smachine and varies only on the master machine. In the following experiments, ran all the workload parameters for more than five iterations. The iteration results are analyzed to identify the efficacy of software services on cloud infrastructure.

5.1 Trust Level Maintenance

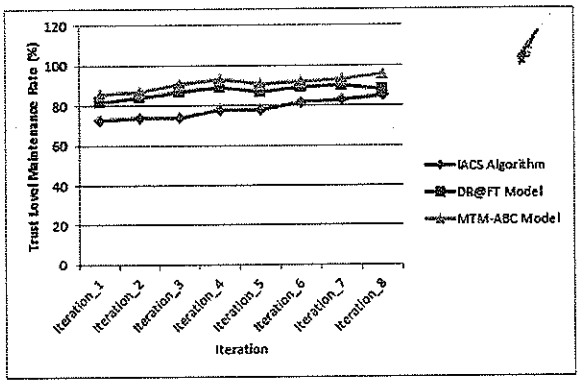
The percentage result on the authentication level helps to identify the trust level of the data transfer from the client to server virtual machine. The trustlevel on the users is also needed in order to choose thetrustworthy software services that are offered by cloud providers.

Iteration	Trust Level Maintenance Rate (%)		
	IACS Algorithm	DR@FT Model	MTM-ABC Model
Iteration_1	73	82	86
Iteration_2	74	84	87
Iteration_3	74	87	91
Iteration_4	78	89	93
Iteration_5	78	87	91
Iteration_6	82	89	92
Iteration_7	83	90	93
Iteration_8	85	88	96

Table 1 : Trust Level Maintenance RateTabulation

Table 1 shows the trust level maintenance rate produced on different set of request iterations. In addition, the trust level maintenance rate on different iterations is compared with the other two methods,IACS Algorithm [1] and DR@FT Model [2] respectively. From the values tabulated in above Fig 1, we can observe that the eight different instances are considered for conducting the experiments using the Cloudsim simulator.

To illustrate the trust level rate on each VM client machine request, tabulated valuesare shown clearly. The request sent to the cloud server VM to perform the process respectively, with high trust rate. The client and server VM within the cloud can share the data with high trust level.The minimum and maximum rate of trust value denotes the percentage of the authentication level. Because of the most unauthorized user access in the cloud model, our proposed system evaluates the trust level maintenance rate parameter. The evaluated proposed work result is compared against the state of arts [1, 2]. In this paper, trust level between the client and server VM are



carried out by using the Amazon.com website information.Amazon web site information is requested from the clients VM and trust is maintained based on the evaluation. Amazon web site request accepted a machine is considered as the VM server.

Figure 4 : Trust Level Maintenance Rate Measure

Fig 4 illustrates the trust level based on iteration. Trust Level maintenance rate as illustrated in Fig 4, increases linearly on different time of iterations. The increase in maintenance rate is justified since the more request from the different VM client are sent to the VM server, and these request are authenticated using the trust function in MTM-ABC Model. The trust function is increased by 12 – 22% in MTM-ABC Model when compared with IACS Algorithm [1] and also increased by 3 – 9 % when compared with the DR@FT Model [2]. The trust level is improved in our research work by computing the Eigen Server Trust Function. The trust function measures the authenticity of the multiple requests (i.e., multi-tenants requests) on the cloud infrastructure VM server.Requests from the different variety of tenants are computed effectively with the trust function in MTM-ABC Model.

5.2 Average Response Time

No. of Client VM Request	Average Response Time to Tenants (ms)		
	IACS Algorithm	DR@FT Model	MTM-ABC Model
5	2000	1620	1480
10	3610	3200	2600
15	5400	5000	4500
20	7300	6700	6040
25	8600	8000	7320
30	9990	9500	8700
35	12700	11600	10300

The average amount of time taken to transfer the software services to the client virtual machines from the server systems is aid for the response time measure. The average response time is reduced, when the client request is not long waited in queue.

Table 2 : Tabulation of Average Response Time to Tenants

Average Response Time = Client Request VM - Response From Server VM

Results for computing the response time based on the VM client requests are given in the table 2. The table describes the response time taken for satisfying the tenant's requests. In above tabulation set of 5, 10, 15...35 requests are taken concurrently, to measure the response time measure. The result of the response time is observed to be linearly increased for the different set of VM client requests. The linear increase in the response time denotes the proposed system is effective when compared with the existing IACS Algorithm [1] and DR@FT Model [2].

In order to observe the response time required for achieving the high performance, the scenario was executed for the seven times with the different VM request count.For the each cloud simulator run, the request

response time is noted to identify the efficiency. As a result, the result of each scenario was computed with an average of seven simulation runs. The lesser waiting time in the queue improves the response rate and efficiency.

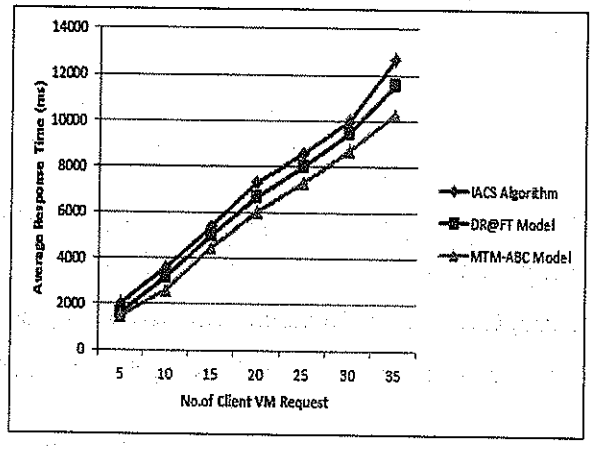


Figure : 5 Measure of Average Response Time to Tenants

Figure 5 compares the average response time of MTM-ABC Model to that of the IACS Algorithm [1] and DR@FT Model [2]. In all the state of affairs, the MTM-ABC Model clearly outperforms the other two systems on measuring the response time. Time taken is lesser when compared with other two existing state of arts. As illustrated, MTM-ABC Model is 12 – 27 % lesser when compared with the IACS Algorithm [1], by suing the history information. The history information is assessed to identify the tenant's interest towards the software services from the cloud infrastructure. The cloud zone produces the services to the VM clients (i.e., tenants) with high trust level security. On the other hand, the MTM-ABC Model is 8 – 18 % lesser response time taken when compared with the DR@FT Model [2]. Dual approach (i.e., client and server) machine start time of process is used to measure the average response time.

5.3 Overall Flexibility

The overall flexibility of the cloud system includes the factors such as authentication level, security and usability.In this study software data are collected from the cloud users and place in the sever machines. The flexibility level is developed in order to provide the software services with the trust value in cloud based applications.

Table : 3 Tabulation of Flexibility Level

Technique	Flexibility Level
IACS Algorithm	85
DR@FT Model	87
MTM-ABC Model	92

The Flexibility level of the MTM-ABC Model, IACS Algorithm, and DR@FT Model are evaluated. The flexibility of the system depends on the rate at which the software services provide to the authorized clients in the high security level. The security level is highly maintained on the cloud software service accessing machines. Table 3 shows the flexibility level over the combination of the three systems [1, 2 and proposed work]. The flexibility is maintained in the high rate even on the different configured VM server and clients.

Figure : 6 Measure of Flexibility Level

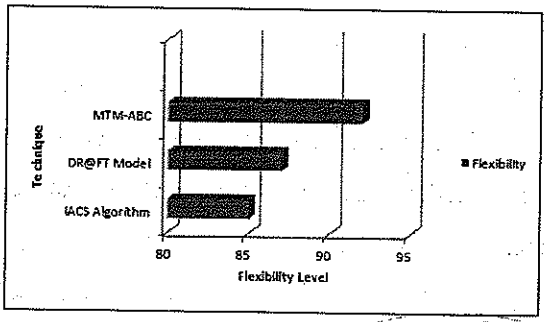


Figure 6 shows the behavior of the flexibility level in response with the security. The flexibility level on the varying count of the VM server requests is evaluated. Flexibility level of MTM-ABC system is increased based on the authentication level, security and usability ratepercentage. The authentication level is higher with larger usability of tenants automatically improves the flexibility rate. Finally, fig 6 shows the higher flexibility rate that is caused by the development of the active Bayes classifier in proposed work. The Bayes classification used for classifying the user request to the products. The similar product request from the multiple tenants is also classified, to improve the flexibility rate. Flexibility level is improvesapproximately by 7 % in MTM-ABC Model when compared with the IACS Algorithm [1].The flexibility rate is improved by 5 % when comparing with DR@FT Model.The tenants request also measures the trust functions to improve the authentication level.

Active Bayes Classifier in MTM-ABC Model produces the high authentication with the independent analysis. The tenants independently measure the trust score value based on the responded software stack.The higher trust score improves the authentication level on cloud software servicing. Effective software services through the proposed system still improve the usability rate. The combination of higher result on the authentication level, security and usability factor helps to improve the flexibility level.

5.4 Throughput Level on accessing the software from cloud zone

Throughput is the amount of work that a cloud systems do in a given time period. Historically, throughput has been a measure of the comparative effectiveness of large commercial software services to the client virtual machines.

Throughput = Rate of software Service * Time rate for servicing

The throughput level on accessing the software from the cloud infrastructure depends on the time rate at which the service is exactly provided. Rate of the software service depends on the speed of transferring the software services.

Table : 4 Tabulation of Throughput Level

Software Service Count	Throughput Level on accessing Software (MB)		
	IACS Algorithm	DR@FT Model	MTM-ABC Model
50	41.06	45.47	50.15
100	25.45	28.15	30.29
150	15.16	18.96	20.68
200	46.56	50.15	55.12
250	61.30	65.23	70.478
300	65.49	71.77	75.41
350	45.73	50.82	55.56
400	71.88	75.10	82.89

Table 1 shows the throughput level on accessing the software from the VM machine. The throughput level measure is carried out on the IACS Algorithm [1], DR@FT Model [2] and MTM-ABC Model. Form the values tabulated we can identify the throughput level of different software count. The services based throughput level on the cloud infrastructure is measured in terms of Mega

Bytes (MB). The increased throughput level improves the system performance under cloudsim simulator.

To estimates the throughput level, the different software services points are taken for the evaluation. The different software services are provided to the clients form the cloud infrastructure. For instance if the 50 different software services are taken for the evaluation, the throughput level is improved by 50.15 percent in MTM-ABC Model, whereas the throughput achieves only 41.06 percent in IACS Algorithm [1], and 45.47 percent in DR@FT Model [2].

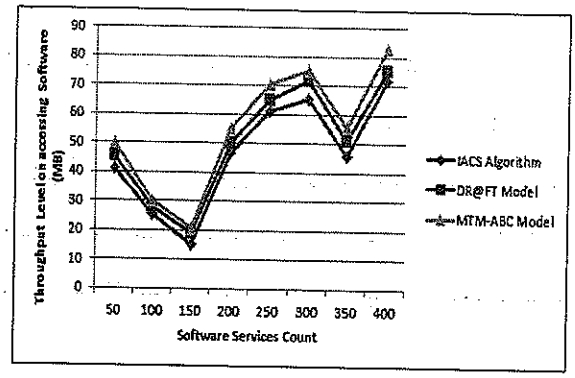


Figure : 7 Measure of Throughput Level

Figure 7 illustrates the throughput level based on the service count. As illustrated in the graph, throughput level varies for the different service count. The increase in throughput level is aid based on the rate of data transfer from the cloud VM server machine. The maximum delivery of software services to the client machines improves the throughput rate by 14 – 36 % in MTM-ABC Model when compared with the IACS Algorithm [1]. Software services sharing to the multiple tenants are high in MTM-ABC Model when compared with the existing state of arts [1, 2] methods.

Similarly, the throughput level of the DR@FT Model [2] is compared with the MTM-ABC Model. The proposed work is 5 – 10 % higher when compared with the DR@FT Model [2] because of the active trust evaluation function on the VM server side. The trust level is evaluated to improve the authentication level on different software service count. On dynamically creating trust assessment based on different situations software services, helps to achieve the high throughput level.

Finally, MTM-ABC model works to self-manage the trust level on dual approach (i.e., client and server side). Active Bayes Classifier in the client model dynamically makes the trust decision in the different situation of the user request to the software services. MTM-ABC model attain the high trust maintenance rate, with minimal response time. The flexibility is also raised with increased throughput on accessing the software services from cloud infrastructure.

VI. Conclusion

In this paper, we have proposed a novel Multi-Tenant Active Bayes Classifier Trust Maintenance (MTM-ABC) model to maintain the trust level on cloud infrastructure software services. It facilitates the dual role and task principle to establish the trust model between the clients and servers. In our model the users are assigned to the security domains that relate with trust functions. The active client trust evaluation function is used in the MTM-

ABC- Client model to compute the authentication level based on services provider through prior knowledge. Simultaneously, Eigen Server Trust Function is provided on the MTM-ABC-Cloud Server model for the authentication to the multiple tenants requests. The model secures the access and flow of data from the cloud infrastructure. Every task of request uses the active Bayes classification to attain the high security level.The level of trust value from the trust function is attained in the MTM-ABC-Cloud Server model to deploy the service model.The perspective of this work, produce the experimental result with the 8.71 % improved throughput rate and 10.76 % minimal response time.The model implemented with high authentication on the cloud zone for providing the software services.

References

[1] Imad M. Abbadi., and AnbangRuan., *"Towards Trustworthy Resource Scheduling in Clouds,"* IEEE Transactions on Information Forensics and Security, Vol.8, No. 6, June 2013.

[2] SmithaSundareswaran., Anna C. Squicciarini., and Dan Lin., *"Ensuring Distributed Accountability for Data Sharing in the Cloud,"* IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 9, NO. 4, JULY/AUGUST 2012

[3] Vijay Varadharajan., UdayaTupakula., *"Counteracting security attacks in virtual machines in the cloud using property based attestation,"* Journal of Network and Computer Applications., Elsevier journal., 2014

[4] Mehdi Sookhak., HamidTalebian.,EjazAhmed., AbdullahGani., Muhammad Khurram Khan., *"A review on remote data auditing in single cloud server: Taxonomy and open issues,"* Journal of Network and Computer Applications., Elsevier journal., 2014

[5] Younis A. Younis., KashifKifayat., MadjidMerabti., *"An access control model for cloud computing,"* Journal of information security and applications, Elsevier journal., 2014

[6] Jorge Bernal Bernabe., Juan M. Marin Perez., Jose M. AlcarazCalero., Felix J. Garcia Clementec., Gregorio Martinez Perez., Antonio F. Gomez Skarmeta., *"Semantic-aware multi-tenancy authorization system for cloud architectures,"* Future Generation Computer Systems., Elsevier journal., 2014

[7] Ajay Basil Varghese., T Hemalatha., SangeethaSasidharan., and ShanyJophin.., *"Trust Assessment Policy Manager in Cloud Computing – Cloud Service Provider's Perspective,"* Int. J. on Recent Trends in Engineering and Technology, Vol. 10, No. 1, Jan 2014

[8] ManjuAsnani., AparnaVerma., Sunil Bijve., BrijeshBakariya., *"Sketch on Security Issues and Concerns of Cloud Environment: Review,"* International Conference on Cloud, Big Data and Trust 2013

[9] Vijay.G.R., Dr.A.Rama Mohan Reddy., *"Investigational Analysis of Security Measures Effectiveness in Cloud Computing: A Study,"* Computer Engineering and Intelligent Systems, ISSN 2222-1719, 2014

[10] Maha TEBAA., Said EL HAJII., *"Secure Cloud Computing through Homomorphic Encryption,"* International Journal of Advancements in Computing Technology(IJACT)., 2013

[11] Sangwook Kim., Hwanju Kim., Joonwon Lee., JinkyuJeong., *"Group-based memory over subscription for virtualized clouds,"* J. Parallel Distrib. Computing, Elsevier journal., 2014

[12] Jiwu Shu., Zhirong Shen., Wei Xue., *"Shield: A stackable secure storage system for file sharing in public storage,"* J. Parallel Distrib. Computing, Elsevier journal., 2014

[13] FaridDaryabar., Ali Dehghantanha., NurIzuraUdzir., Nor Fazlidabinti Mohd Sani., Solahuddin bin Shamsuddin., FarhoodNorouzizadeh., *"Survey About Impacts of Cloud Computing on Digital Forensics,"* International Journal of Cyber Security and Digital Forensics., 2013

[14] Kui Xu., HuijunXiong., Chehai Wu., DeianStefan., and Danfeng Yao., *"Data-Provenance Verification For Secure Hosts,"* IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 9, NO. 2, MARCH/APRIL 2012

[15] DimitriosZissis., DimitriosLekkas., *"Addressing cloud computing security issues,"* Future Generation Computer Systems., Elsevier Journal., 2012

AUTHORS BIOGRAPHY

Chidambaram K, Tamilnadu, India, 10 June 1982. I received Bachelor of Science (B.Sc) in Physics Regular from Sri Ramakrishna Mission Vidyalaya College of Art and Science, Coimbatore, Tamilnadu, India in 2002. I completed Master of Computer Applications (M.C.A) in Computer Application (Regular) from Kongu Engineering College, Perundurai, Tamilnadu, India in 2005. I am having 9 years of experience in Software Industry as a SOA Tester and now playing the Test Lead Role in Wipro Technology.

Dr.C.Chandrasekar, Tamilnadu, India, 16 May 1972. He received Master of Computer Applications (MCA) from Madurai Kamarajar University, Madurai, Tamilnadu, India and Doctor of Philosophy (PhD) from Periyar University, Salem, Tamilnadu, India. He has 17 years of experience in both Teaching and research. Currently he is working as Associate Professor in Department of Computer Science from Periyar University, Tamilnadu, India. Previously he worked as Assistant Professor in K.S.Rangasamy College of Technology.