ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

An ISO 3297: 2007 Certified Organization Vol.5, Special Issue 2, April 2017

An International Conference on Recent Trends in IT Innovations - Tec'afe 2017

Organized by

Dept. of Computer Science, Garden City University, Bangalore-560049, India

Enhancing Fault Tolerance of Virtual Machine In Cloud

Shameer Basha K.R, Vikash Kumar

Dept. of Master of Computer Applications, JAIN University, Bangalore, India

Dept. of Computer Science & IT, JAIN University, Bangalore, India

ABSTRACT: Cloud Computing is a way of Computing wherein service is given over the internet by applying different models. Fault Tolerance is a huge support to ensure availability and reliability of major services in executing applications. Here we are proposing model to study how the systems tolerates the faults and makes a conclusion on the reliability of the nodes that are being processed i.e. VM (Virtual Machine).If a VM leads to a result within a given time, its reliability automatically increases and if it doesn't meet the expectation within the stipulated time the reliability decreases. If the VM persistently fails, it will be separated and a new VM will be connected. There is also a lowest reliability level. If any VM does not reach the expectations, the system will conduct backward recovery or protection. The proposed techniques are established on designing various variants on many VM, and assigning reliability to the conclusion given by the variants. The instances of a VM can be of similar or non-identical types. The system gives both the forward and backward mechanism but the main emphasis we have is on Forward Recovery.

KEYWORDS*:* Cloud; formatting,FT(fault tolerance), VM(virtual Machine); Reliabilty

I. INTRODUCTION

Cloud computing allocates the resources like networking, storages and physical storages. Theygive many services through their cloud service providers. The famous one's currently ruling the market are Amazon, windows Azure, Google and many more,every service provider has his own unique services to offer depending on the need of the users or clients. We can see that Google the giant provides all available cloud services like PaaS,IaaS and Saas. The cloud computing is constructed on the distributed idea and it very helpful for all the users. This paper focuses on the research done of fault tolerance in the cloud. In a cloud territory, there are lots of hidden or concealed nodes known as VM (virtual machine).VM is effectively a program that is installed to run virtually not physically. In a server, it is a processing machine. In the cloudterritory, the data of a user is replicated in numerous VM's. The user demand is sent to all accessible VM's.

If a given node or a VM fails, then it will not reply and all the accessible VM will reply to the demand that has been made by the user. The fault tolerance weighed should distinguish the Best reliable VM's among the lot and reply to the user demand. This paper is used to find or discover the reliable VM's in the cloud territory

II. FAULT TOLERANCE AND ITS ENCOUNTERS

A. Researching FT(Fault Tolerance)

FT (Fault Tolerance) is a huge concern for guaranteeing availability and reliability of important services and also execution of applications. FT (Fault Tolerance) is a productive means to achieve reliability. It also means that the computer should be up and running even when it might be subjected to fault presence.

B. Faults encountered

There are a number of faults that have been encountered. Few techniques/policies have been developed and proposed to address this issue

Fault Tolerance Reaction [7] : Reactive FT or Fault tolerance reaction policies lowers the consequences faced by the faults during execution of application when the faults are occurring persuasively. Multiple techniques that provide solution like Retry, Replay etc .

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

An ISO 3297: 2007 Certified Organization Vol.5, Special Issue 2, April 2017

An International Conference on Recent Trends in IT Innovations - Tec'afe 2017

Organized by

Dept. of Computer Science, Garden City University, Bangalore-560049, India

Predetermining FT [7] : Predetermining FT is to completely avoid a recovery well before the faults can even occur by anticipating them and replace the deduced components by functional components. The multiple solutions are Migrating preventive, S/W Rejuvenation etc.

III. RELATED WORK

Studies have been carried out in this field of cloud infrastructure with regards to fault tolerance. But there are many things that are yet to be discovered in fault tolerance of VM based cloud infrastructure.

New cases have been introduced related to fault tolerance.

A. *WENBING ZHAO et al. (2010) presented a new idea Low Latency Fault Tolerance(LLFT) that follows leader follow approach in the cloud environment for the distributed applications by providing fault tolerance.*

B. *ANJU BALA et al. (2014) submitted an idea to design a smart fault detections by predetermining the task failures.*

C. *SHIVAM NAGPAL et al (2013) suggested a FT model thattakes decisions. Usually the reliability of a node is gauged at on 2 things : time and accuracy. If either one of it fails at certain given level then the system is scheduled to perform backward recovery*

IV. PROPOSED WORK

Our proposed model is formed on improving FT (Fault Tolerance) and assessing the reliability of VM (Virtual Machine) in cloud based environments. This method uses two types of nodes. One is VM set and adjudication node such as main node. The VM applies acceptance test for the logical validity. Time checker, decision mechanism and reliability assessor algorithm are used by the adjudicator node to detect the reliable VM. The reliable VM is distinguished and used to process the client request. The data of the VM is in a compressed state that the client can accept. A VM is chosen for computation on premise of least/minimum and Maximum reliability. The maximum reliability produced by a node is determined as the system event output. Fault Tolerance can be provided by storing the data on numerous cloud using the virtualization techniques

- Acceptance Test (AT) [2]:This module scans if there is a chance of fault occurring. Host failure is the fault. The AT acknowledges both failure and success of the VM. The failed VM is not taken into consideration. If the data within the VM gets corrupted it will not affect the VM so this VM is reviewed by the TC. The result is only passed to the TC module when it is successful. The AT sends a verify signal to TC to indicate the failure.

- Time Checker (TC) [2]: It is estimated in milliseconds. It is given by upper and lower bound limit. The response time and monitoring time is given in milliseconds. During failure TC module executes recovery.

- Reliability Assessor (RA) [2]: The assessing of reliability for each module is seen here. Through main core module the reliability is found for the proposed system, for the improvement of reliability of VM we need to check which system tolerated the faults and takes decision based on processing nodes of the reliable systems. All the VMs are 100% reliable at the beginning of each computing cycle. If a node is able to achieve the desired result within a given time then its reliability increases by using RF (reliability Factor) . We can see the decrease of reliability when the result is failed to be produced in the given time using adaptability factor m.

  - Algorithm 1:

Begin

Initially rel:= 1, m:= 1

Input from config RF, maxRel , minRel

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

An ISO 3297: 2007 Certified Organization Vol.5, Special Issue 2, April 2017

An International Conference on Recent Trends in IT Innovations - Tec'afe 2017

Organized by

Dept. of Computer Science, Garden City University, Bangalore-560049, India

Input nodestatus

if nodestatus =Pass then

*rel :=rel+(rel*RF)*

if m> 1then

m:= m-1;

else if nodeStatus =Fail then

*reliability := reliability - (rel*RF*m)*

m:=m+1;

if rel>= maxRel then

rel:= maxRel

if rel<minRel then

nodestatus :=dead

call_proc : remove_this_node

call_proc: add_new_node

End

- DECISION MECHANISM (DM)[1]: Locates the Reliable VM found through resource availability and recent history.
- RECENT HISTORY: It holds the checkpoints. Each computing cycle has checkpoint in it. Backward recovery is done when all the nodes fail, this is usually done with the help of checkpoints kept.
- RESOURCE AVAILABILITY: Memory is counted as a resource. Every VM has its own memory availability.
  - Algorithm 2:

Begin

Initially rel=1,m=1

Input from RA nodeRel, Numvand nodes

Input from configuration SRL

bestRel1 :=find_reliability of node with highest reliability

bestRel2 :=find_reliability of node with second most reliability

bestRel3:=find_reliability of node with third most reliability

if bestRel1&&bestRel2&&bestRel3>=SRL

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

An ISO 3297: 2007 Certified Organization Vol.5, Special Issue 2, April 2017

An International Conference on Recent Trends in IT Innovations - Tec'afe 2017

Organized by

Dept. of Computer Science, Garden City University, Bangalore-560049, India

status :=success

else

perform_backward_recovery

call_proc: remove_node_minRel

call_proc: add_new_node

End

The initial value is given as 1. Adaptability factor is taken as m, which commands the adaptability of reliability assessment. The m value should always be greater than 0. Input of three factors are taken into consideration minReliability(minimum reliability), maxReliability(maximum reliability) and RF(Reliability factor). The RF is the factor that either increases or decreases the node's reliability. When a node attains minReliability level, it is halted from moving any further operations that may take place. maxReliability is in the beginning of a situation the desired output is seen in succeeding cycles but fails every time. Hence the reliability should not be more than necessary required to lower the reliability. For example, we have two nodes both have a passing and failing score of 5 each in 10 cycles. but the node, which has more failures recently has high chance of being less reliable than the other node. The variables such as (minReliability, maxReliability, SRL, RF) depend on the application value.

V. SCOPE OF STUDY

As per the research there is high possibility to the need of implementing unmanned FT.Academicians conducted a literature review and found many challenges that need to overcome in the future.[3]

- Chances of error occurrence is more as the processing is done on a remote based computer.
- Since cloud environment is scalable, unpredictable it is difficult to interpret the changing system state.
- The user has less information because of the high system complexity, hence that makes it difficult to plan an optimal fault tolerance result.
- The diverse nature of cloud is it biggest drawback to localize the faults. A technique needs to be found to locate faults.

V. EXPERIMENTAL SETUP

We have configured the VM as below

Table-1: VM CONFIG

PARAMETER	VALUE
Architecture	X86
OS	WINDOWS
VMM	XEN
COST	2.0
COSTPERMEM	0.04
COSTPERSTORAGE	0.001
COSTPERBW	0.0

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

An ISO 3297: 2007 Certified Organization Vol.5, Special Issue 2, April 2017

An International Conference on Recent Trends in IT Innovations - Tec'afe 2017

Organized by

Dept. of Computer Science, Garden City University, Bangalore-560049, India

Table-2: HOST CONFIG

PARAMETER	VALUE
RAM	1000
STORAGE	10000
BW	10000

Table-3: DATACENTER CONFIG

PARAMETER	VALUE
Architecture	X86
OS	WINDOWS
STORAGE COST $/S	0.1
DATA TRANSFER COST $/Gb	0.1
PHYSICAL UNITS NUMBERS	2

Data is being sent to adjudication node to the VM and results are being received. The VM takes 1 millisecond for lower limit and 1.5 for upper limit. All VM execute the algorithm at the same time.

VII. CONCLUSION AND FUTURE WORK

Cloud environment is ever changing and scalable due to which unforeseen behavior can be seen leading to failures and faults. To improve the reliability the failures should be assessed and taken care of conclusively. By proposing this model we are trying to make our systems more fult tolerable than and we are also taking the dynamic aspect of a cloud environment to our advantage. The decision making algorith finds the three top most reliable systems witrh the highest reliable ratio amongst many. The likeliness of failure is very small in our work. Due to this forwrd recoveryu takes place till the nodes fail to give the expected outcome. Due to Decision Mechanism the backward recovery is performed if the given nodes are unable to achieve the desired System Reliability Level (SRL) .Since checkpointing is made when the desired result is gained through all the nodes it does not have a domino effect. Due to this process the QOS of cloud is vastly improved since it is very efficient as we find the three top reliable systems. When conducted a test between VM's the top 3 reliable VM had the following output VM1- 0.74 , VM2- 0.70, VM3-0.74. The other VM had 0.69,0.64,0.65 etc. These numbers were tested against 1 as max reliable outcome.
We are working on some improvemental techniques that can be applied and to make our system the least fault tolerant as possible , In future major focusing will be done in decision mechanism .

REFERENCES

[1] Sambare, A. S., Meshram, A.D., & Zade, S.D (2013). Fault Tolerance Model For Reliable Cloud Computing.
[2] Rajesh.S & Kanniga Devi, Improving Fault Tolerance in VM Based Cloud Infrastructure. Journal of Innovative Research in Science,Engineering and Technology. ISSN Online(2319-8753) Print(2347-6710)
[3] Zeeshan Amin, Nisha Sethi & Harshpreet Singh, Review on Fault Tolerance Techniques In Cloud Computing. International Journal Of Computer Applictions (0975 - 8887)
[4] M .J. Rehman, S. Malik, Time Stamped Fault Tolerance inDistributed Systems. IEEE International Multitopic Conference, Karachi, Pakistan.
[5] L. L. Pullum, Software Fault Tolerance and Implementation- Artech House, Boston, London, United Kingdom, 2001
[6] Mohammad Jahangir Alam, Galam Moktader Nayeem - Analysis of various Software Fault Tolerane Techniques, 2006
[7] Fault Toelrance- Challenges, Techniques and Implementation in Cloud Computing - Anju Bala, Inderveer Chana.
[8] D. Caromel, C. Delbe, A. D. Costanzo, M. Leyton, ProActive: An Integrated Platform for Programming and Running Applications on Grids and P2P Systems, Computational Methods in Science and Technology 12(1), pp 69-77, 2006

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

An ISO 3297: 2007 Certified Organization Vol.5, Special Issue 2, April 2017

An International Conference on Recent Trends in IT Innovations - Tec'afe 2017

Organized by

Dept. of Computer Science, Garden City University, Bangalore-560049, India

[9] K. H. Kim, Towards Integration of Major Design Techniques for Real-Time Fault-Tolerant Computer System, Society for Design & Process Science, USA, 2002
[10] O. S. Unsal, I. Koren, M. Krishna, Towards Energy-Aware Software Based Fault Tolerance in Real Time Systems ISLPED '02, Monterey, California, USA, August 12-14, 2002
[11] Tchana Alain et. al. Fault Tolerant Approaches in Cloud Computing Infrastructures. The Eight International Conference on Autonomic and Autonomous Systems (2012): 42-48.
[12] M.Armbrust, A.Fox, R. Griffit,et al., A view of cloud computing , Communications of the ACM, vol. 53, no. 4, pp. 50–58, 2010.
[13] Webbing Zhao et. al. Fault Tolerance Middleware for cloud computing.Third International Conference on Cloud Computing (2010): 67-74.
[14] Gang Chen, Hai Jin, Deqing Zou, Bing Bing Zhou, Weizhong Qiang, Gang Hu, SHelp: Automatic Selfhealing for Multiple Application Instances in a Virtual Machine Environment , IEEE International Conferenceon Cluster Computing, 2010
[15] L. M. Vaquero, L. Rodero-Merino, J. Caceres and M. Lindner, A break in the clouds: towards a cloud definition, SIGCOMM Computer Communication Review,vol. 39, pp. 50–55, December 2008.