THINK INDIA (Quarterly Journal)

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

A Deep Insight Into Fault Tolerance In Cloud Computing

[1]J. Florence Mary, [2]C.V.Akshaya, [3]S. Gomathi

[1,2] I MSc Data Analysis, [3]Assistant Professor

Department of Computer Science (PG)

PSGR Krishnammal College for women

Coimbatore

ABSTRACT

Cloud computing is an emerging computing technology that provides users with the capabilities of it as a service and allows access to these services on the internet without having specialized information or controlling infrastructure. Fault is the one of the biggest challenges in making a system fault tolerant. To reduce the impact of failure on the system and application execution ,failures should be anticipated and proactively handled. Fault tolerance is used to predict the failures and take a proper action before failure occurs. Cloud virtualized system architecture using Haproxy has been proposed. In the proposed system automatic fault tolerance has been implemented. The results shows that the proposed system can deal with various software faults for server applications in a cloud virtualized environment.

INTRODUCTION

Cloud computing is a concept that refers to applications and services which are run on a distributed network with the help of available resources. Cloud storage not only provides us the massive computing infrastructure but also the economics of scale. This needs guarantee of the quality of data storage services which involves two concerns from both cloud users and cloud service providers: data integrity and storage efficiency [1]. The main advantage of cloud computing is to provide reliability, low cost, high availability, scalability and flexibility for end users which appears as a new computing paradigm [2]. Service quality also plays an important role in cloud computing. To reach the needed performance and consistency, we need fault-tolerant computer environment, which has

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

the ability to tolerate faults by isolate defect modules, detecting failures due to that the rest of the environment can operate correctly. When one of the server goes down while multiple instances of an application are running on several virtual machines ,there is a need to implement an autonomic fault tolerance technique that can handle these types of faults. To address this problem, cloud virtualized system architecture has been proposed and implemented using HAProxy. In order to attain robustness in cloud computing, failure should be determined and handled carefully. In this paper we will give basic knowledge about Fault tolerance Approaches, the Methods used for fault management in cloud we also study some existing fault management models which tolerates fault in cloud environment.

TYPES OF FAULTS

The faults can be classified based on several factors.

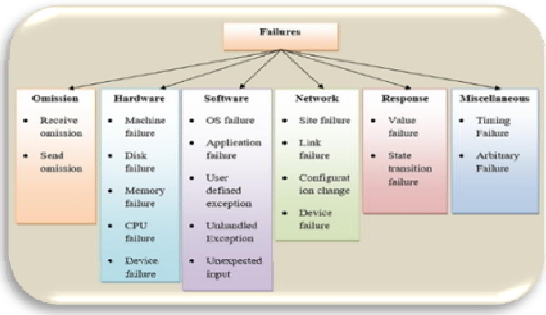


Fig.1.1 Classification of faults[10]

- **Network fault** : A Fault that occurs in a network due to network partition, Congestion, packet loss, packet corruption, destination failure, link failure., etc.
- **Physical faults** : This Fault can occur in hardware like fault in CPUs, Fault in memory, Fault in storage, Power failure etc.
- **Media faults** : Fault that occurs due to media head crashes in storage media.
- **Process faults** : A fault which occurs due to shortage of resource, software bugs, inefficient processing capabilities etc.

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

- **Service expiry fault** :When an application is using a resource ,the service time of a resource may expire.A fault can be classified on the basis of computing resources and time.
- **Permanent failure** : These failures occur by accidental power breakdowns, natural disasters., etc. These failures can cause main disruptions and some part of the system may not be working as wanted.
- **Intermittent failure** : These failures happen occasionally. These failures happen while the system performs its operations. The damage caused due to these failures is very difficult to be identified.
- **Transient failure** : These failures are caused by some inherent fault in the system and they can be easily resolved. These failure can be corrected by Retrying /Roll back the system to previous state. These failures are common in the computer system.

TYPES OF FAULT TOLERANCE

The fault tolerance is majorly classified in to two types. They are hardware and software fault tolerance.

Hardware fault tolerance : Most of the computer systems are directed automatically to recover from failure that occurs due to hardware components. Each and every component is backed up with protective redundancy, so that if one component fails the other can be alternated to perform the functions with any failure. Example: Mirroring technique which recovers the failure of storage media. The general hardware fault tolerance approaches are fault masking and dynamic recovery.

Fault Masking : Fault masking is a structural redundancy technique that completely masks faults within a set of redundant components. A number of identical components execute the same functions, and their outputs are voted to remove errors created by a faulty module. Triple modular redundancy (TMR) is a commonly used form of fault masking in which the Component is triplicated and voted. The voting process for selecting the redundant component can also be triplicated so that individual voter failures can also be corrected by the voting process. A TMR approach fails whenever two components in a redundant triplet failed so that the vote is no longer valid.[8]

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

Dynamic recovery : Dynamic recovery technique is used only when one copy of the task or computation is made to run at a time. This technique performs self-repair. As in fault masking technique, additional spare components are used to perform back up operation. (Proactive redundancy). In this dynamic recovery technique a special recovery mechanism are required to detect faults in module. Instead of switching to redundant component the fault in the module or the component is analyzed and recovered from them using the actions like rollback, initialization, retry, and restart. These actions perform the restore operation and continue computations.

Software fault tolerance : The faults in the software can be tolerated by using static and dynamic approaches similar to the hardware fault tolerance approaches. N-version programming uses static redundancy in the form of independent written programs that perform the same operation. In an N-version programming, each module is made with up to N different implementations. Each programmer accomplishes the same task, but hopefully in a different way. Each version then submits its answer to voter or decider which determines the correct answer and returns that as the result of the module. If the acceptance test fails, a redundant code block is executed. Design diversity approach combines both of hardware and software fault tolerance by implementing a fault tolerant computer system using different hardware and software in redundant channels [7].

Fault tolerance techniques can be classified into types depending on the policies of fault tolerance via,

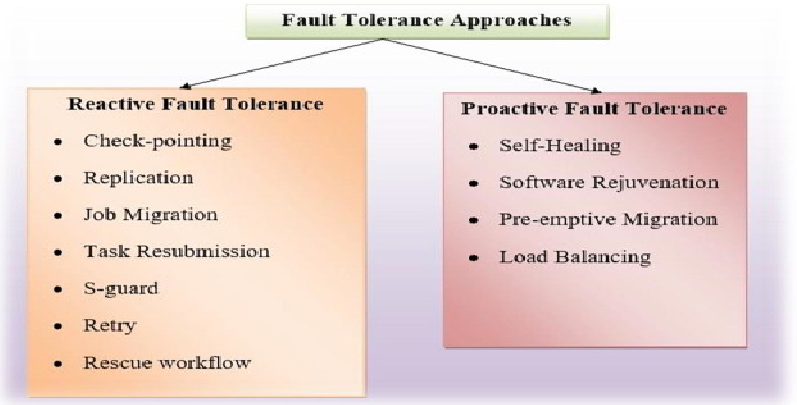


Fig 1.2 Classification of fault tolerance techniques[10]

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

Proactive Fault Tolerance:Proactive fault tolerance means early prediction of the problem before it actually arises.

Reactive fault Tolerance:This policy handles the failure. When the failure actually occurs, the effect is reduced. This could be further divided into two sub-procedures:

1. Error Processing
2. Flaw Treatment

The first process eliminates error from the system. Fault treatment prevents faults from getting reactivated.

Fault tolerance is accomplished by error processing. Error Processing has two main phases. The first phase gives more importance on effective error processing, which means bringing the effective error back to a latent state and if possible it is done before occurrence of a failure. The Second Phase is latent error processing which aims to ensure that the error is not reactivated

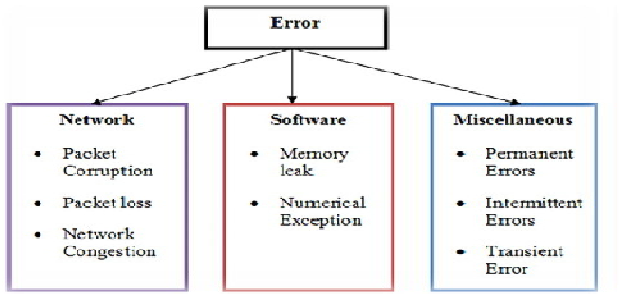


Fig 1.3 Types of errors[10]

Existing Fault Tolerance Approaches in Cloud

The various techniques that are used for fault tolerance in cloud are:

Check pointing : It is a good fault tolerance approach .It is used for applications which have a long running time. In check pointing technique , check pointing is done after each change in system state. It is useful when a task is not able to complete. It fails in the middle due to some error. Then the task is made to begin from the most recent check pointed state instead of restarting it from the beginning.

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

Task Migration : There are some cases when a task in not able to complete on the assigned specific virtual machine . When this failure occurs then that task could be moved other machine. This can be performed by using HA-Proxy.

Self- Healing : Big tasks can divided into parts .This is done for better performance. It results in creation of variant application instance. The instances run on distinct virtual machines, like this automated failure management is done for instances.

Safety bag checks : This strategy is quite simple. These blocks the command which does not met the requirements for safe execution or proper working of machine.

S-Guard : It is a stream processing techniques. It makes available more resources. It uses the mechanism of Rollback recovery. Check Pointing is done asynchronously. It is used for distributed environment. S-Guard is performed using Hadoop or Amazon EC2.

Task Resubmission : Task failure can make the complete job to miscarry. So when a failed task is recognized ,it should be submitted to same or either different resource for reexecution.

Rescue workflow : This method is used for Fault tolerance in workflow execution.

Reconfiguration : The configuration of the system is changed in this technique. The faulty component is removed.

Resource Co-allocation : It increases the availability of resources. It takes care of multiple resources. Resource allocation is performed to complete the execution of task.

Table 1: Comparison of various fault tolerance tools[11]

Fault Tolerance Techniques	**Policies**	**System**	**Programming Framework**	**Environment**	**Fault detected**	**Application on type**
Self-Healing, job migration, Replication	Reactive/ Proactive	HAProxy	Java	Virtual machines	Process/node failures	Load balancing fault tolerance

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

Check pointing	Reactive	SHelp	SQL,JAVA	Virtual machines	Application failures	Fault tolerance
Check pointing, Retry, Self-Healing	Reactive/ Proactive	Assure	JAVA	Virtual machines	Host/Network failures	Fault tolerance
Job Migration, Replication, S-Guard	Reactive/ Proactive	Hadoop	Java,HTML,CSS	Virtual machines	Application/node failures	Data intensive
Replication, S-Guard, Task Resubmission	Reactive/ Proactive	Amazon EC2	Amazon Machine Image, Amazon Map	Virtual machines	Application/node failures	Load balancing and fault tolerance

Fault Tolerance Models

Fault Tolerance Models are designed using these techniques.

AFTRC : A Fault Tolerance Model for Real Time Cloud Computing is designed by keeping the fact in mind that real time systems have good computation. These systems are also scalable and make use of virtualization techniques which helps in executing real time applications more effectively. This model is designed by considering the dependability issue. The model makes use of proactive fault strategy and predicts the faulty nodes [3].

LLFT : Low Latency Fault Tolerance act as a middleware for tolerating faults [4]. It is very useful for distributed application which is running in cloud. In this, fault tolerance is provided like a service by cloud providers. Applications are replicated by middleware. In this way replication helps in handling of faults for different applications.

FTWS: Fault Tolerant Workflow Scheduling is a model based on replication approach [5]. It also makes use of resubmission technique. A metric is maintained for checking the priority of tasks and they are submitted accordingly. Principle of workflow is used in this model.

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

Workflow means a series of task executed orderly. When the workflow is scheduled fault management is done.

FTM : It is one of the most flexible model. It delivers fault tolerance as on demand service. The user has an advantage that without having known the working of model ,they can specify the required fault tolerance [6]. It is mainly designed for dependability issues. It consists of various components.

Candy : It is component base availability modeling frame work. It is mainly designed for availability issues. System modeling language is used to construct a model from specifications. This is done semi automatically.

Vega-warden : It is a uniform user management system. This creates a global work space for variant applications and distinct infrastructure. This model is specially constructed for virtual cluster base cloud computing environment to overcome the 2 problems: usability and security which arise from sharing of infrastructure.

FT-Cloud : It has a mechanism of automatic detection of faults. In this model frequency is used for finding out the component.

Magi-Cube : It is a kind of architecture for computing in cloud environment. It is designed for dependability, expenditure and performance issues. All three issues are related to storage. This architecture provides highly reliable and less redundant storage. This storage system is done for metadata handling. It also handles file read and write.

Proposed cloud environment

In this research paper we are extending this work into the cloud environment with the help of some experts tool because cloud environment is a difficult one. Shelp system can continuously work even when the software application fault occur virtual machine environment [9]. Cloud is in the requirement of automatic fault tolerance and process switching from one server to another server. Due to that requirement we are installing HAproxy in one server to monitor the application process running in the all the servers in cloud environment properly. If any one of the server get fails HAproxy switch the application process into another server for uninterrupted service [10].

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

we will take three servers and proxy server. Among that server 1 & 2 are application server and Server 3 is backup server. Server has been installed Linux OS and SQL database. HAProxy server is used to handle server failures in fault tolerant cloud environment. Configured environment will support java and SQL.

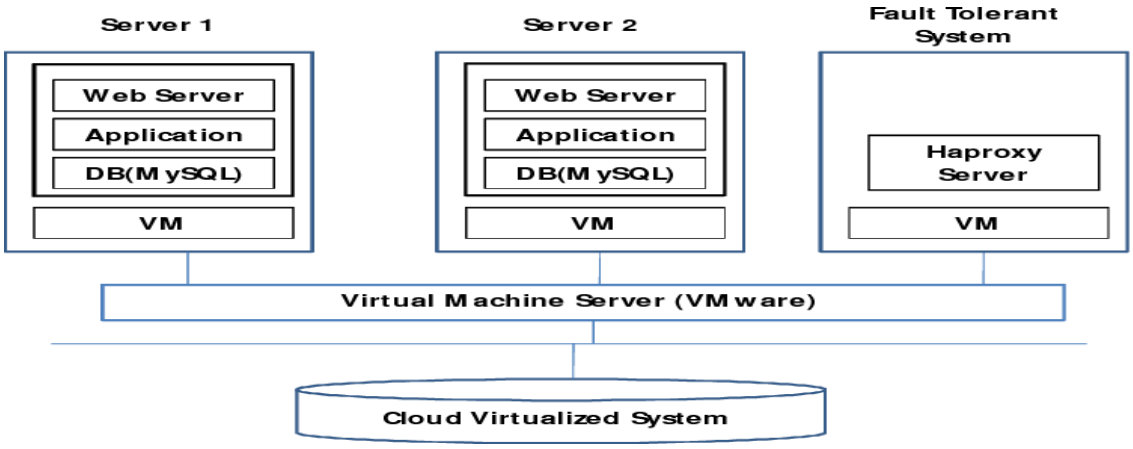
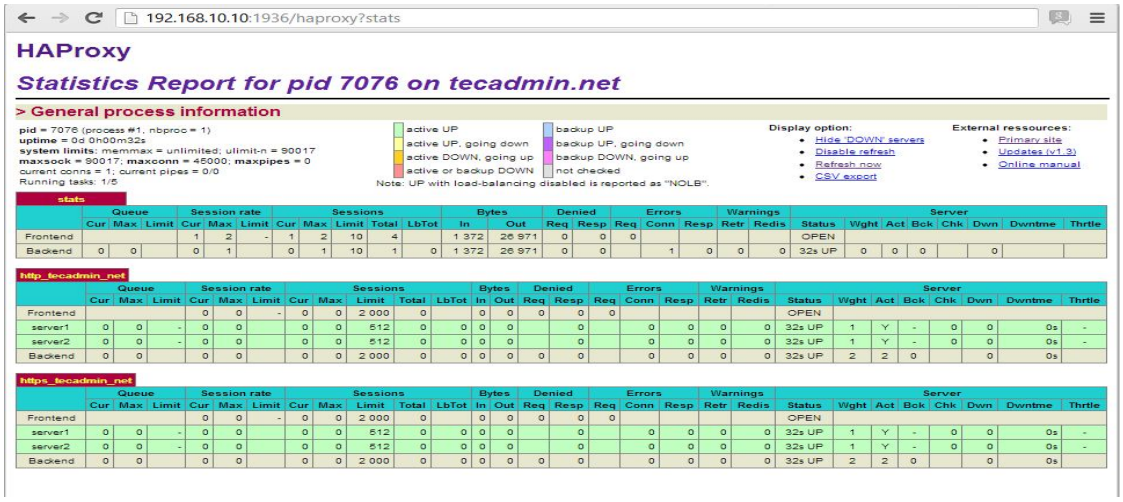


Fig 1.4 Cloud virtualized system environment[12]



HAProxystatistics[14]

CONCLUSION

Fault tolerance is about tolerating, avoiding or improving faults and errors remained in the system after its development and implementation. This paper discussed the types of faults and

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

various types of fault tolerance techniques and comparison of various fault tolerance techniques.andalso discussed about the various tools used for fault tolerance and their comparison.

The concept of fault tolerance is concerned with all techniques necessary to enable a system to tolerate software faults remaining in system after its development. A cloud virtualized system architecture is also proposed using HAproxy. Automatic fault tolerance is also implemented dealing with various faults for server application in cloud virtualized environment. The connection will be automatically redirected to other server when one of the servers goes down.

REFERENCES

[1] Himanshu Agarwal, Anju Sharma, "A Comprehensive Survey of Fault Tolerance Techniques in Cloud Computing", International Conference on Computing and Network Communications (CoCoNet), IEEE, 2016.

[2] K. V. Vishwanath and N. Nagappan, "Characterizing cloud computing hardware reliability," in Proceedings of the1st ACM symposium on Cloud computing, ACM, pp. 193 – 204, New York, USA, 2010.

[3] Sudha Lakshmi, S. "Fault Tolerance in Cloud Computing", ACICE, Vol.04, Special Issue 01, 2013.

[4] Zhao, W., Melliar P.M. and Mose, L.E." Fault Tolerance Middleware for Cloud Computing" 2010 IEEE 3rd International Conference on Cloud Computing. DOI: 10.1109/CLOUD.2010.26, 5-10 July 2010, p. 67 -74.

[5] Kaur, J., Kinger, S. "Analysis of Different Techniques Used For Fault Tolerance", IJCSIT: International Journal of Computers & Technology, Vol.4 - Issue 2, (2013), 737-741

[6]B. S. Taheri, M. G. Arani and M. Maeen, "ACCFLA: Access Control in Cloud Federation using Learning Automata", International Journal of Computer Applications 2014.

[7]2010 IEEE 3rd International Conference on Cloud Computing.

[8] Gang Chen, Hai Jin, DeqingZou, Bing Bing Zhou, WeizhongQiang, Gang Hu, "SHelp: Automatic Selfhealing for Multiple Application Instances in a Virtual Machine Environment", IEEE International Conference on Cluster Computing, 2010.

ISSN:0971-1260

Vol-22-Issue-4-October-December-2019

[9] Ravi Jhawar, Vincenzo Piuri, Marco Santambrogio, "A ComprehensiveConceptual System-Level Approach to Fault Tolerance in Cloud Computing 2012

IEEE

[10] Journal of King Saud University – Computer and Information Sciences (2018).

[11] International Journal of Database Theory and application (2015).

[12]International journal of advances in engineering science and technology

[13]A project on autonomic fault tolerance using HAProxy in cloud environment

