International Journal of Computer Architecture and Mobility (ISSN 2319-9229)
Volume 7 -Issue 1, January 2019

A Comparative Study Of Fault Tolerance Techniques In Cloud Computing

Mohit kumar sahu
mohit9479sahu@gmail.com

Abstract- Cloud computing is a method of computing in a place that provides users with the capabilities of information technology as a service and allows them to have access to these services on the Internet without having specialized information or controlling the infrastructure. Due to the exponential growth of cloud computing, the need for fault tolerance in the cloud is a key factor for investigation. Fault tolerance has all the necessary techniques to keep active power and reliability. The main advantages of using fault tolerance in cloud computing include failure recovery, lower costs, and improved performance criteria. The motivation to examine existing techniques and models of fault tolerance in cloud computing has encouraged researchers to participate in the development of more efficient algorithms. So, in recent years, there has been a lot of research on fault tolerant systems. This has led to new strategies to find out the benefits and barriers of fault tolerant systems in cloud computing. This paper introduces various fault-tolerance methods recently used in cloud computing and a comprehensive review is presented of various techniques and models in the field of error tolerance in cloud computing. By studying and analyzing fault-tolerant techniques, according to the requirements and criteria we can use the perfect fit for error tolerance in cloud computing.

1. **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 computing when software and applications are run provides an abstract representation of physical systems
    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 in which the efficiency and reliability of the service can be considered as two important aspects.
    3. Each of performance and reliability play an important role in cloud computing, because if the service reliability is low, it occasion frequent crashes in the cloud service, which in turn results in a reduction in the number of customers and in the result is a loss for the server. If the reliability of the service is not high, but its efficiency is low, users who ask for services should wait for a long time and this will also be a disappointment to them. So it can be directly related to tolerance to error. Tolerance to fault is used In order to improve reliability in cloud computing. Fault tolerance is one of the important issues in cloud computing and is related to all the necessary techniques to enable the system to tolerate the remaining software fault in the system after its development.
    4. Techniques provide reliability and validity in the cloud environment. The main advantages of the implementation of the fault tolerance technique in cloud computing are: failure recovery, low cost, improved performance criteria, and so on.

    In this paper, we study the techniques of error tolerance in cloud computing and the key stimulation for this study is to identify different models and methods of fault tolerance in cloud computing which will support researchers in creating a more efficient algorithm.

2. **Fault tolerance -** Fault tolerance is a feature of the system that prevents a computer system or network device from failing due to any fault or failures in system execution. The fault tolerance includes effective steps to prevent such errors or failures in the system. In fact, a fault-tolerant system is capable of providing the service in question in an efficient manner if one or more faults or failures occur in system components and the availability

and reliability will not be lost of the system. A tolerable bug system is a form that can In the event of a bug, tolerate it and continue to work. Perhaps it's better to have a definition of error at first, because with this word, two words are also mentioned in the mind that is the fault and the failure. But there are three differences between them.

Failure: A failure occurs when an expected system is not functioning correctly so, if the system misconduct affords the system to fail at least one of its capabilities properly, then the system is in a malfunction. Fault: The cause of the failure is a fault in the system. So the fault is a physical malfunction or a failure of a hardware or software component.

Bug: The Afford of an error is a bug in the system "Not necessarily all bugs will not lead to faults" Failure, faults and bugs may occur in applications, virtual machines, and even hardware. The system must be capable of handling the fault and continue to operate.

Fault detection: To provide each evaluation, the first step that a system must perform is identifying the fault functions.

Fault Repair: After the system detects a fault, the next step is to avoid the fault or to improve it.

Fig - The path to failure

3. Types of fault tolerance

Fault tolerance can be classified in two categories of hardware fault tolerance and software fault tolerance.

3.1 Hardware fault tolerance

One of the main goals of fault tolerance is make the computer system which can automatically recover if multiple random faults occur in hardware components. The developed methods for this work generally include the partitioning of a computational system in several modules. Each module in the system has been redundant Therefore, if the failure occurs in one of the modules, the backup

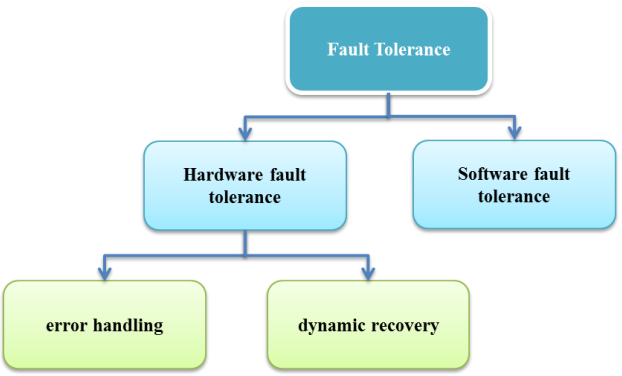

Fig - Types of Fault Tolerance

module will continue to work. Fault tolerant methods include two types of error handling and dynamic recovery.

3.2 Software fault tolerance

Software faults (programming faults) can be exploited using static and dynamic methods similar to those used for hardware fault handling. One of these methods is n-version programming, which uses static redundancy in the form of independent programs. All of them are doing the same thing. There is another method called design variation, which incorporated

software and hardware fault tolerance by applying a fault-tolerant computer system using hardware and software in redundant channels [8].the main target of the design diversity technique is to tolerate hardware and software faults, but the cost is very expensive.

4. Fault-tolerant requirements

An important goal in designing distributed systems is to create a system in such a way that it can automatically repair minor defects without generally affecting the overall system performance. In particular, whenever a malfunction happened, the system must be repaired in an acceptable action. In other words, a distributed system is expected to tolerate the fault. To realize the role of tolerable fault, there should be covered a number of useful requirements for distributed systems including the following

Average Failover Time (MTTF): The waiting time for failure bearing in mind that the system has been utilizable.

Average Repair Time (MTTR): This mentioned to the expected time to repair the system after a failure.

Average Breakdown Time (MTBF): This represents the average time for the next failure and is calculated as follows:

$$MTBF=MTTF+MTTR$$

Reliability: Points to the point that a system can run continuously and without fail.

$$Reliability = \frac{MTTF}{1+MTTF}$$

Availability: The system operates at any one time and is available to perform tasks.

$$Availability = \frac{MTBF}{1+MTBF}$$

Safety: refers to a situation when nothing works temporarily, nothing happens.

Maintenance: refers to how to repair a failed system. A superb maintenance system may also have a high degree of availability.

5. Fault Tolerance Policies in C$Maintainability = \frac{1}{1+MTTR}$

Based on fault tolerance, different techniques and strategies are classified as follows:

5.1 Tolerable Preventive fault

The policy of tolerance is the action-oriented fault, preventing faults and failure by their prediction and the actionist replaces the dubious component. It detects the problem before it happens. Some techniques based on this policy contain preventive migration, software rejuvenation, self-healing, and so on.

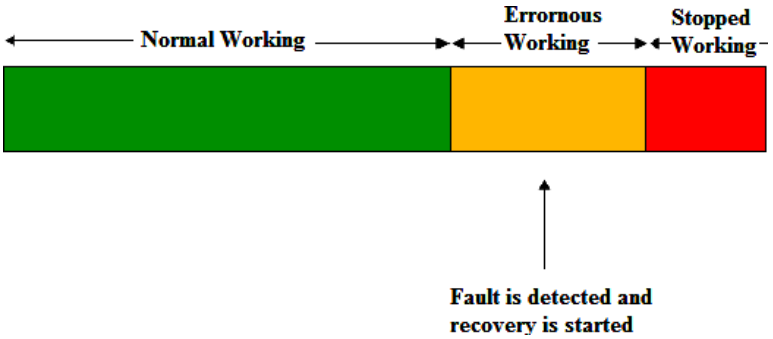

Fig - Timetable for a precautionary fault detection system

5.2 Tolerable reaction fault

Responsive fault tolerance policy, when failure happens effectively attempt to reduce failures. This method creates more power for the system. A number of different techniques are based on this policy; Inspection / restart, re-assignment of function, release of workflow, handling of user's specific behavior, retrial, labor migration, S-Guard, etc.

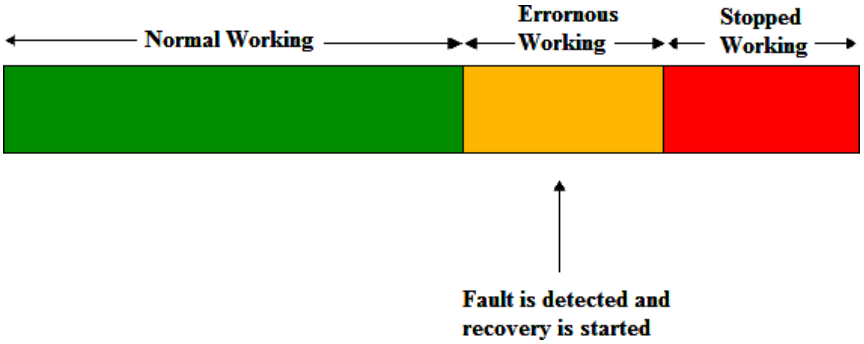

fig - Timeline for a reactive fault detection system

5.3 Adaptive fault tolerance

The tolerance of an application fault requires the change depending on the range of control inputs and current position in its space. Adaptive tolerance automatically adjusts the instructions to the status control. And ensuring the credibility of critical modules under any resources and time constraints Provides as much redirection resources and modules as possible.

Table (1): Comparison of different tolerance techniques

Politics	System	Programming framework	Environment	Fault detection	Application type	Techniques
Preventive / Reactionary	HAProxy	Java	Virtual machine	Node failure or processing	Fault tolerance and load balancing	**Self-healing, job migration, replication**
Preventive	SHelp	SQL,JAVA	Virtual machine	Application crash	Fault tolerance	**Check Pointing**
Preventive / Reactionary	Assure	JAVA	Virtual machine	Network / host failure	Fault tolerance	**Check Pointing, retry, self-healing**
Preventive / Reactionary	Hadoop	Java,HTML, CSS	Cloudy environment	Node downtime and application programs های کاربردی	Volume of data	**Job Migration, Replication, S-Guard**
Preventive / Reactionary	Amazon EC2	Amazon Machine Image, Amazon Map	Cloudy environment	Node downtime and application programs	Fault tolerance and load balancing	**Replication, S-Guard, Task Resubmission**

6. Fault tolerant models in cloud computing

Nowadays, various models are presented based on the techniques mentioned in the previous section. Most of them are based on these techniques.

FT-Cloud: Cloud computing applications are usually provided on a large scale and with complexity. However, unfortunately, their reliability is still far from ideal. In reference it is a chart-based ranking component which uses its architecture to build cloud applications. It is a two-stage framework of operational algorithms that includes ranking and fault tolerance. FT-Cloud provides cloud computing applications to deal with faults. This fault-tolerant model offers cloud computing against collision faults and amounts.

BFT-Cloud: Repeat policy is used in this model. This model can be categorized as a reactive model. When a request is located in the cloud computing system, the request must be performed in different nodes.

LLFT model: The model introduced in, which is called LLFT, delivers fault tolerance with a low postponement. This model has provided F.T the capacity to develop distributed programs or data centers. LLFT uses the leader / follower duplication approach.

Candy model: In reference, has been introduced a component-based distributed modeling framework which is a useful and comprehensive semi-automatic model and is described by the language of the system model. This model indicates that cloud services and cloud computing providers should guaranteed availability, which is one of the highlights of cloud services.

AFTRC Model: A fault tolerance model for cloud computing is based on a real-time system can benefit from computing capacity and cloud-scalable virtual environment for better real-time use. In this model, the fault system is tolerant of action .Based on the reliability of the processor nodes; it exports an executable command.

FTWS model: The FTWS model stated in, fault tolerance is presented using startup and propagation techniques according to the priority of tasks. This model plans the workflow with a deadline in the presence of faults. This model is also based on the fact that the workflow is a set of processed tasks with a segregated order based on data and affiliation.

Magi-Cube Model: In, the Magi-Cube architecture is introduced which a very reliable storage structure for cloud computing. They use this system on top of HDFS and handle it as a storage system for reading / writing files and managing cloud data.

Table (2): Comparison of policies applied in each of the models

Tolerable models for cloud computing	Policies and techniques				
	Preventive		Reaction		
	Self-repair	Preventive Migration	Inspection	Duplicate	Labor migration
BFT-Cloud	×	×	×	✓	×
FTM	×	×	✓	✓	✓
LLFT	×	×	×	✓	×
FTWS	×	×	✓	✓	×
Candy	×	×	×	✓	×
Magi Cube	×	×	×	✓	✓
AFTRC	×	×	✓	✓	✓

7. Conclusion

Cloud computing has become a commonly used computing technology, which very popular. There must be reliability and availability for users. This requires utilize of tested tolerance methods which can manage any kind of fault in every aspect. The fault tolerance is required when a fault enters the system borderline, Therefore, fault tolerance techniques are used to predict these failures and take the necessary actions before the damage happened. Reliability and availability are two important parameters in cloud computing. Therefore, we need a fault tolerance method that will prepare the services provided in cloud computing against the resulting faults and failures. in this paper, we discussed the need for fault tolerance, covering various techniques for implementing fault tolerance. There are a number of fault-tolerant techniques in the cloud, which prepare various fault tolerance mechanisms by increasing the reliability of the system. Also, these techniques represent a major role in providing service availability to the user. But there are still some issues and deficiencies that must be considered for each framework. There are debility that none of them can complete all aspects of the faults. Therefore, it is possible to dominate the debility of all previous models and try to create a proper and efficient model that covers the most aspects of fault tolerance. In the future, it is also expected to better understand the types of faults in hardware, software, and cloud infrastructure by providing other models of architecture with higher fault tolerance, higher reliability, availability, and more impressive performance.

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] Bo Yang, F. Tan and Y.S. Dai, "Performance Evaluation of Cloud Service Considering Fault Recovery," The Journal of Supercomputing, vol. 65, no. 1, pp. 426-444, 2013.

[4] V.M.Sivagami, K.S.EaswaraKumar, "Survey on Fault Tolerance Techniques in Cloud Computing Environment", International Journal of Scientific Engineering and Applied Science (IJSEAS) - Volume-1, Issue-9, December 2015.

[5] Pankaj Deep Kaur, Kanu Priya, ".Fault Tolerance Techniques and Architectures in Cloud Computing - A Comparative Analysis", International Conference on Green Computing and Internet of Things (ICGCIoT), 2015.

[6] Abhishek Bhavsar, Ameya More, "A Holistic Approach to Autonomic Self-Healing Distributed Computing System" International Journal of Computer Applications (0975 – 8887) Volume 76– No.3, August 2013.

[7] HARPREET KAUR, AMRITPAL KAUR, "A SURVEY ON FAULT TOLERANCE TECHNIQUES IN CLOUD COMPUTING", International Journal of Science, Engineering and Technology, Volume 3 Issue 2, 2015.

[8] Amritpal Singh, Supriya Kinger, "An Efficient Fault Tolerance Mechanism Based on Moving Averages Algorithm" © 2013, IJARCSSE, ISSN: 2277 128X .

[9] Salma M. A. Ataallah, Salwa M. Nassar, "Fault Tolerance in Cloud Computing – Survey", 11th International Computer Engineering Conference (ICENCO), IEEE, 2016.

[10] Bala.A, Chana.I.(2012),"Fault Tolerance- Challenges, Techniques and Implementation in Cloud Computing" International Journal of Computer Science Issues