ISSN: 2348-4098
Volume 3 Issue 2, 2015

A SURVEY ON FAULT TOLERANCE TECHNIQUES IN CLOUD COMPUTING

[1]HARPREET KAUR, [2]AMRITPAL KAUR

[1]Mtech Research Scholar, SGGSWU, Fatehgarh Sahib, Punjab, India, Email: grewal.harpreet55@yahoo.in

[2]Assistant Professor, SGGSWU, Fatehgarh Sahib, Punjab, India, Email: ap.kaur89@gmail.com

ABSTRACT

Cloud computing becomes very popular these days and fault tolerance is a major issue in this cloud environment. This paper is dedicated to study different kinds of faults and various techniques for handling them. This survey mainly focuses on why fault tolerance is required in cloud computing and which techniques are used to handle these failures so that services can be delivered without delay and applications will execute without failure. By taking an appropriate action before or after the occurrence of failure, the system can be predicted.

General Terms-*Cloud Computing, Fault Tolerance, Reactive, Proactive*

1. INTRODUCTION

Cloud computing is most popular in both industry and academia. Cloud computing has different definitions but all have a main focus on these key characteristics: the ability to deliver IT services at lower value and risk however the critical services become more reliable and available to the user. [1]

In this computing paradigm there are modern data centers and in these data centers there are thousand servers internally connected to each other. Several applications are hosted on these servers. Most often, these large data centers are not real while virtual, and the computing resources are provided to the user over the Internet in the form of configurable Virtual Machines (VMs). By this technique, several resources are provided to the individual user [2].Cloud computing provides many good services to the user. Now a day the user level is highly increased to utilize the services. In cloud computing there are various problem areas and fault tolerance is one of them. Fault tolerance is necessary for the system to guarantee both availability and reliability of critical services and application execution. To minimize the occurrence of failures in the system and its impact on application execution, failures should be handled by suitable technique. The survey purpose of existing fault tolerance techniques in cloud computing is to study the shortcomings of these existing techniques and developing new algorithm for handling fault in a better way. This paper discusses about various aspects of faults and the need for fault tolerance in cloud computing. The rest of the paper has following sections. Section 2 discusses the requirement of fault tolerance in cloud computing. Section 3 describes types of faults and section 4 discusses techniques of fault tolerance in cloud computing.

2. REQUIREMENT OF FAULT TOLERANCE IN CLOUD COMPUTING

Although cloud computing has gained more popularity in the industry, still several research problems associated with various areas like fault tolerance, workflow scheduling, load equalization, security etc are currently present. Currently fault tolerance is a major issue in cloud computing. All existing techniques related to fault tolerance need to be enhanced to handle fault specifically. When a fault occurs in the system there should be such techniques that can take appropriate action according to the type of fault. [3]

There are various fault avoidance and fault removal techniques but these are not enough in cloud computing. The system should be fault tolerable to attain higher reliability. Faults have to be compelled to be handled either proactively or reactively i.e. either before or once the incidence of actual fault whereas the system continues to deliver services acceptable to users. This goal can be achieved by the employment of various algorithms like error detection algorithm, recovery algorithm, migration algorithm, and fault diagnosis. The fault tolerance is needed in cloud computing to extend reliability, availability to a good extent so that the services can be delivered without long delay and wastage of resources and energy.

3. TYPES OF FAULTS

Cloud platform has 3 types of layers: hardware, VM's and applications. Each of them worries with failures. The failures will be there in any layer either in hardware or VM, throughout the execution of application. So according to the nature of fault appropriate action is taken. These faults will be classified based on many factors like [4]:

3.1 NETWORK FAULTS

ISSN: 2348-4098
Volume 3 Issue 2, 2015

Faults related to the network are known as network faults. These faults occur when data does not reaches the destination due to various reasons like destination unreachable, link broken, corrupted packet etc is known as network fault.

3.2 PHYSICAL FAULTS

Physical faults are related to hardware such as CPUs faults, storage faults, memory faults etc.

3.3 MEDIA FAULTS

Fault occurs due to media head crashes known as media faults.

3.4 PROCESSOR FAULTS

When operating system crashes or any event like that happens then its called fault occurs and known as processor fault.

3.5 PROCESS FAULTS

When due to any reason resources becomes shorter or any software bug appears then its called that process fault occurs.

3.6 SERVICE EXPIRY FAULTS

When a situation comes that any resource's service time expires but then also the application continues to use it. This type of fault is called service expiry fault.

Faults may also be classified as [5]:

3.7 TRANSIENT

This type of fault occurs for some time duration and appears only once and then disappears after taking an action. For example, at first attempt the network message can't reach its destination but upon retransmission after some time it successfully reaches the same destination.

3.8 INTERMITTENT

Intermittent faults are those that are reoccurring again and again. These faults are very irritated type and occur mainly due to either the failure of any component or improper operation between the components, for example a faulty connection.

3.9 PERMANENT

This type of failure persists within the system until the systems faulty components either repaired or in some cases fully replaced. Examples of these faults are disk crashes, errors in software etc.

4. EXISTING FAULT TOLERANCE TECHNIQUES

There are different types of faults which can occur in cloud computing. Various fault tolerance techniques are used. By using these techniques fault can be handled either proactively or reactively.

4.1 REACTIVE FAULT TOLERANCE

When actual fault occurs in the system, then the reactive fault tolerance policies help to reduce the impact of failures on the execution of application. The fault is detected after it enters in the system and starts the recovery henceforth. These techniques though provide good fault tolerant solution for general computing environment, cannot fulfill the time constraints set by real time computing systems.

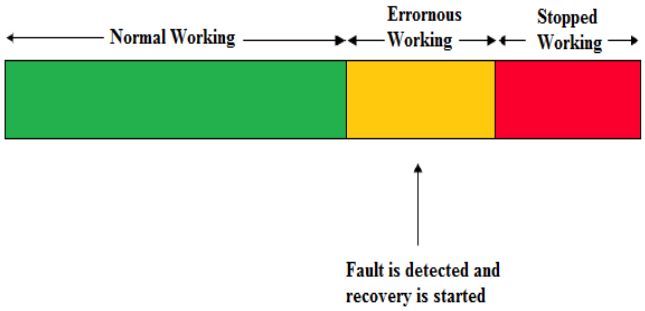

Figure 1: Timeline for a Reactive fault detection system

Some reactive fault tolerance techniques are as follows. [3]

4.1.1 Checkpointing/Restart

When doing task scheduling the checkpoints are inserted to identify fault incidence. These techniques take less computation and less time as a result of the task is restarted at the previous checked point. There is no ought to restart the full task. [6]

4.1.2 Replication

Replication is a very effective technique in fault tolerance. In this technique, there are various replicas of any task on different resources. When any fault occurs then execution continues to succeed until all replicated tasks are destroyed. [7]

4.1.3 Job Migration

When due to resource failure or machine failure any task fails then the task is shifted on another virtual machine where it continues its execution.

4.1.4 SGuard

It is less tumultuous to normal stream process and makes a lot of resources out there. SGuard is predicated on rollback recovery and might be enforced in HADOOP, Amazon EC2.

4.1.5 Task Resubmission

Task resubmission means that when any task fails then it is recommitted either to the same machine or a different one. [7]

4.1.6 User defined exception handling

When any task/job fails then the user gives the efficient treatment to that failed jobs for workflows.

ISSN: 2348-4098
Volume 3 Issue 2, 2015

4.1.7 Rescue Workflow

In this technique, the workflow continues even if the task fails until it becomes impossible to move forward without catering the failed task.

4.1.8 Job Retry (JRT) Strategy

In this strategy first of all the unsuccessful task is re-executed on the similar resource from starting. If the task continues to fail on different executing points, then to get rid of the additional overhead the average execution time is computed. A threshold is additionally set to limit the number of retries of the failed job on the same machine.

4.2 PROACTIVE FAULT TOLERANCE

The principle of proactive fault tolerance policies is to avoid extra effort for recovering the failed job, nodes, by predicting failures in advance and proactively replace the suspected components with other working components. Proactive fault tolerance systems are able to fulfill the time constraints set by the real time systems.

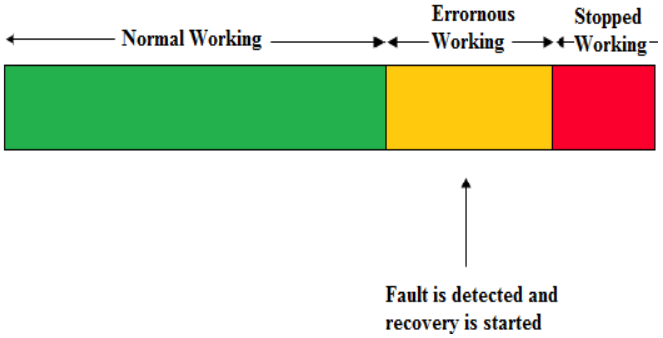

Figure 2: Timeline for a Proactive fault tolerant system

Some proactive fault tolerance techniques are as follows [3].

4.2.1 Software Rejuvenation

Software Rejuvenation is the method in which an application is immediately terminated and then restarted as a clean state. There are various rejuvenation intervals and the application is restarted at every interval with a clean internal state. [8]

4.2.2 Using Self Healing

When multiple instances of a single application are running on different multiple virtual machines and when fault occurs by using self healing the failure of different application instances can be handled automatically.

4.2.3 Using Preemptive Migration

By using this technique, the parts of an application running on a computing node that is about to fail are migrated to a different nodes. By using so the application is migrated to different node before actual fault occurs. [9]

Based on these different techniques, there are several fault tolerance models are implemented.

"AFTRC" A fault tolerance model for real time cloud computing. This is a model in which, the system tolerates faults and makes the choice on the premise of dependability of the process nodes, i.e. virtual machines. The dependability of the virtual machines is adaptive that changes when each computing cycle. If a virtual machine manages to provide an accurate result within the time limit, then it becomes more reliable and if it fails to provide the result within time or incorrect result, then its reliability decreases. During this planned model the system tolerates the fault proactively and makes the decision on the premise of dependability of the process nodes. [10]

"FTM" is a model to overcome the limitation of the prevailing methodologies of on-demand service. In order to have greater reliability and resiliency they propose an innovative perspective on making and managing fault tolerance model. By this methodology user will specify and apply the desired level of fault tolerance while not requiring any knowledge regarding its implementation. FTM architecture thus can be viewed as a collection of many web service components, each having a specific functionality. [11]

"LLFT" The Low Latency Fault Tolerance (LLFT) middleware provides fault tolerance for distributed applications deployed in cloud computing or data centers environment, using the leader or follower replication approach. In LLFT there is minimum end-to-end delay and maintains robust replica consistency. The replicas states consistency is very robust, in each case when it's free from fault and when there is a fault. When the fault appears, the LLFT reconfiguration or recovery mechanisms ensure that there is a backup. The state is transferred from the existing replica to the new replica and operations of both are synchronized. [12]

"FTWS" Fault Tolerant Workflow Scheduling algorithm provides fault tolerance by using replication and resubmission of tasks based on priority of the tasks. The replication of tasks depends on a heuristic metric which is calculated by finding the tradeoff between the replication factor and resubmission factor. FTWS schedules workflows within a deadline even in the presence of failures without using any history of information. The experiments were conducted in a simulated cloud environment by scheduling workflows in the presence of failures, which are generated randomly. The experimental results of the proposed work demonstrate the effective success rate in-spite of various failures. [13]

"Candy" A component-based availability modeling framework named Candy, that constructs a comprehensive availability model semi-automatically from system delineated by Systems Modeling Language(SysML). This model is predicated on the very fact that prime availability assurance of cloud service is one in all the most characteristic of cloud service and also one of the main critical and challenging issues for cloud service provider. [14]

ISSN: 2348-4098
Volume 3 Issue 2, 2015

"FT-Cloud" FT-Cloud that is a component ranking based mostly framework for building fault-tolerant cloud applications to attack the challenge of providing extremely reliable cloud applications. FTCloud employs component invocation structures and therefore the invocation frequencies to spot the many parts in a very cloud application. An efficient algorithm is designed to automatically determine optimal fault tolerance strategy for these vital components. [15]

*"**Magi-Cube**"* an extremely reliable and low redundancy storage design for cloud computing. The system is created at the top of HDFS and uses it as a storage system for read and writes files and metadata management. The file scripting and different repair parts are designed to figure within the background. The actual fact behind this model depends is that there are three conflicting components that are minimum cost, better performance, and greater reliability. Magi cube is designed to provide these facilities to a particular model. [16]

"Vega-warden" Vega Warden, a uniform user management system to solve two problems: usability and security. Vega Warden provides a global space to users for different virtual infrastructures and application services in one Cloud, and allows users to interconnect different systems within a homogeneous cloud. This is a uniform authentication model that enables the security isolation of different domains, and the scalability is ensured by the decentralized architecture. [17]

5. RELATED WORK

Ravi et. al. [18] introduced new, system-level approach for creating and managing fault tolerance in clouds. The proposed approach is comprehensive high-level to the implementation details of the fault tolerance techniques for application developers and users by means of a dedicated service layer. Furthermore, he present a scheme that: (I) delivers a comprehensive fault tolerance solution to user's applications by combining selected fault tolerance mechanisms, and (ii) ascertains the properties of a fault tolerance solution by means of runtime monitoring. Based on the proposed approach, he designed a framework that easily integrates with existing Cloud infrastructure and facilitates a third party in offering fault tolerance as a service.

Anjali D. Meshram et al. [19] designed a system that deals with the fault tolerance in cloud computing. In this system, a model named fault tolerance model for cloud (FTMC) based upon the concept of reliability of computing nodes referred to as virtual machines (VM) in cloud environment and the efficient fault tolerance for real time applications running on these virtual machines. The selection of virtual machine for computation depends upon its reliability and might be removed, if it doesn't perform well for real time applications because real time applications are very sensitive to time delay due to the occurrence of fault.

Patra and Singh [20] approached the need of fault tolerance in cloud computing and various proposed models for fault tolerance are discussed and compared on the basis of type of fault and procedure applied for handling fault. Presently, there is a number of fault tolerance models which provide different fault tolerance mechanisms to enhance the system still also multiple challenges resides in every model. So these models strength need to be enhanced.

Tchana et al. [21] have analyzed the implementation of fault tolerance in an exceedingly complicated cloud environment with a spotlight on repair. by analysis its comes to know that in most of existing approaches, both provider and customer can manage the fault tolerance due to which improper or inefficient solution are made, while the solution generated by handling the faults automatically are much more promising. This discussion described by experiments where the autonomic cloud infrastructure is prototyped and the resulted solutions are exclusive and collaborative.

Anju Bala et. al. [3] discusses the prevailing fault tolerance techniques in cloud computing based on the policy of fault, tools used for handling fault and various research challenges. The virtualized architecture is proposed in cloud. In the proposed system the fault tolerance which is implemented is autonomic. The experimental results show that the planned system will trout out various software faults for server applications in a virtualized cloud framework.

Malik et. al. [10] designed a fault tolerance model for real time applications in cloud computing Adaptive Fault Tolerance in Real-time Cloud computing (AFTRC) model is proposed in this paper. This model is highly tolerable to faults and makes the decision on the basis of reliability of the processing nodes. The reliability of machines changes after each computing cycle. There is a time limit set and if any machine provides efficient and correct result within this limit then its reliability increases. But if any machine fails to produce the result within time or produces incorrect result, then the reliability of system decreases. Two recovery methods are provided by the system forward and backward.

6. CONCLUSION

When the fault enters the system boundaries then the fault tolerance methods are required. So the fault tolerance techniques are used to predict these failures and take appropriate action before actually occur. This paper discusses about the need for fault tolerance by covering with its various techniques for implementing fault tolerance. A several number of fault tolerance models present in cloud which provide different fault tolerance mechanisms to enhance the system reliability then also a number of challenges need to be concerned for cloud framework or model. No one of them can fulfill all aspects of fault tolerance. So it is possible to overcome the drawbacks of all previous models and try to make a compact model which will cover maximum fault tolerance aspect.

ISSN: 2348-4098
Volume 3 Issue 2, 2015

REFERENCES

[1] Kunwadee, Sambit Sahu, Yaoping Ruan, Anees Shaikh, and Chitra Dorai Dorai "Are Clouds Ready for Large Distributed Applications? IBM T.J.Watson Research center.

[2] Ravi Jhawar, Vincenzo Piuri, Marco Santambrogioy," A Comprehensive Conceptual System-Level approach to Fault Tolerance in Cloud Computing", DOl 10.1109/SysCon.2012.6189503.

[3] Anju Bala, Inderveer Chana," Fault Tolerance Challenges, Techniques and Implementation in Cloud Computing" IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, No 1, January 2012

[4] Dr. Lakshmi Prasad Saikia, Yumnam Langlen Devi, "Fault Tolerance Techniques and Algorithms In Cloud Computing" International Journal of Computer Science & Communication Networks, Vol 4(1), 01-08

[5] 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

[6] G.Malathy, Dr.Rm. Somasundaram, Dr.K.Duraiswamy, "Checkpoint-Based Fault Identification in Cloud Computing Tasks" International Journal of Engineering Research & Technology (IJERT) Vol. 1 Issue 8, October 2012.

[7] Shivam Choudhary," Task Migration Based Fault Tolerant Scheme for Cloud Computing" international journal of scientific progress and research (ijspr) ISSN: 2349 – 4689 Volume-02, Number- 01, 2014

[8] Y. Huang, C. Kintala, N. Kolettis, and N. Fulton, "Software rejuvenation: Analysis, module and applications," in Proceedings of Fault-Tolerant Computing Symposium FTCS-25, june 1995

[9] Engelmann, G. R. Vall´ee, T. Naughton, and S. L.Scott. Proactive fault tolerance using preemptive migration. In Euromicro International Conference on Parallel, Distributed, and network-based Processing (PDP), pages 252–257, 2009.

[10] Sheheryar Malik, Fabrice Huet, "Adaptive Fault Tolerance in Real Time Cloud Computing" IEEE World Congress on Services, pp 280-287, 2011

[11] Ravi Jhawar, Vincenzo Piuri and Marco Santambrogio"A Comprehensive Conceptual System level Approach to Fault Tolerance in Cloud Computing" IEEE

[12] Wenbing Zhao, P.M. Melliar and L.E. Mose" Fault Tolerance Middleware for Cloud Computing" 2010 IEEE 3rd International Conference on Cloud Computing.

[13] Jayadivya S K, JayaNirmala S, Mary SairaBhanus"Fault Tolerance Workflow Scheduling Based on Replication and Resubmission of Tasks in Cloud Computing" International Journal on Computer Science and Engineering (IJCSE)

[14] Fumio Machida, Ermeson Andrade, Dong SeongKim and Kishor S. Trivedi"Candy: Component-based Availability Modeling Framework for Cloud Service Management Using Sys-ML" 2011 30th IEEE International Symposium on Reliable Distributed Systems.

[15] ZibinZheng, Tom Chao Zhou, Michel R. Lyu, and Irwin king "FT-Cloud: A Component Ranking Framework for Fault-Tolerant Cloud Applications "2010 IEEE 21st International Symposium on Software Reliability Engineering.

[16] QingqingFeng, Jizhong Han, Yun Gao, Dan Meng"Magi-cube: High Reliability and Low Redundancy Storage Architecture for Cloud Computing" 2012 IEEE Seventh International Conference on Networking, Architecture, and Storage.

[17] Jianlin, Xiaoyi Lu, Lin Yu, YongqiangZou and Li Zha"Vega Warden: A Uniform User Management System for Cloud Applications "2010 Fifth IEEE International Conference on Networking, Architecture, and Storage.

[18] Ravi Jhawar, Vincenzo Piuri and Marco Santambrogio, Member of IEEE, "Fault Tolerance Management in Cloud Computing: A System-Level Perspective ", IEEE, 2012.

[19] Anjali D. Meshram, A.S.Sambare and S. D. Zade, "Fault Tolerance Model for Reliable Cloud Computing ", International Journal on Recent and Innovation Trends in Computing and Communication, ISSN 2321 – 8169, Vol. 1, Issue: 7, July 2013.

[20] Prasenjit Kumar Patra, Harshpreet Singh & Gurpreet Singh, "Fault Tolerance Techniques and Comparative Implementation in Cloud Computing" International Journal of Computer Applications, Vol. 64, No.14, February 2013.

A. Tchana and L. Broto and D. Hagimont, "Approaches to Cloud Computing Fault Tolerance," Computer, Information and Telecommunication Systems (CITS), pp 1-6, 2012.