THINK INDIA JOURNAL

ISSN: 0971-1260
Vol-22-Issue-17-September-2019

Future of Fault Tolerance in Cloud Computing

Sarabjit Kumar
Assistant Professor
School of Computer Science & Engineering
Lovely Professional University
Punjab, 144401, India
email id: - sarab.jitkumar@lpu.co.in

Dr. Ajay Shriram Kushwaha
Assistant Professor
School of Computer Science & Engineering
Lovely Professional University
Punjab, 144401, India
email id: - ajay.21908@lpu.co.in

Abstract - Cloud computing have emerged as the IT miracle of 21st century leading to reduction in cost of infrastructure, multifold shoot up in work efficiency, centrally managing the things, interoperability of formats, robust access to storage, increased scope of usability of services irrespective of time and locations and most dominating is minimum dependency cloud resources as services from certain hardware's. Still with lots of advantages the uninterrupted usage of these resources is critical due to machine and human generated exceptions. Which paved a way to decide how should a cloud can respond to these exceptions so that all cloud-based resources can work in fault tolerated way.

Keyword – Cloud computing, inter-operability, uninterrupted, exceptions.

I – Introduction

As per the NIST institute cloud computing is an IT environment which is demand-based and provide easy access model through a network for sharing set of customizable computing resources (like servers, networks, storage, applications, and other services) [1]. The allocation and releasing of these resources are like that is needs minimal effort and cost.

Usually the cloud providing access to its resources to all user are called as public clouds, clouds providing access to limited resources to a specific set of users are called as private clouds, sometime cloud resources are being shared on the basis of similarity like people working on same project or deployed at a particular location are termed as community clouds but sometimes we can counter the mix of these cloud which is usually termed as hybrid clouds [1].

These all are virtual in nature and are represented in most abstracted form to its users providing least scope of customization to its users on architectural components. So, the fault tolerance is also very crucial which provide capabilities to cloud to detect, identify and recover itself from the fault and improve its performance with respect to final output expected by the user.

II – Faults in Clouds and their categories

A failure represents a situation which leads to unexpected result or behavior from a system rather than its required functionality or the expected behavior [2]. Ability of a system to perform its

ISSN: 0971-1260
Vol-22-Issue-17-September-2019

intended functions even a fault exist makes it fault tolerant. Which actually needs that system must understand clearly the existence of error, initiates the corrective actions so that expected result will not get suffered.
Failures can happen at level of servers [3], networks [4], services etc. We can categorize failures occurring in clouds as below: -

a) Failure can be result of crashing of physical devices which is termed as Hardware failure [5] like machine failure, CPU failure, disk failure etc. Services provider of cloud are responsible to respond in such scenarios.

b) Crashing of a logical component usually residing on a Virtual Machine is termed as VM Failure [5]. These types of failure can be dealt by service provider as well as user himself also depending upon the access or authorization to resources he is having.

c) If a user performing some task on cloud and suddenly, he gets unexpected response from the application it results in Application Failure [5]. In such scenario the corrective steps can be carried out by either at service provider level or at the user also.

d) When multiple tasks get crashed due to some logical dependency leading to failure of entire job it results in Job Failure [5].

e) When any of the task get crashed but other tasks continue to work and outcome might get impacted. The task got crashed are reinitiated to overcome the specific challenges. These shows existence of Task Failure [5]

On the basis of the nature of the fault we can address them as

a) A fault can be result of some temporary condition occurring in the system which may halt the system temporarily like connection issue and non-operational service issue in a network. We term these faults as Transient faults [6].

b) A fault can show exhibit features like it may take place randomly at regular interval. There is no permanent diagnose or resolution for such faults. We address them as Intermittent faults [6].

c) A fault which usually continue to exist until its root cause have not been removed. They are addressed as Permanent Faults [6].

d) Some time it is very difficult to detect existence of fault in a system. These faults are the mostly costly on part of there removal and diagnose. We address them as Byzantine faults [6].

III – Effective Fault Tolerance Mechanism

Fault tolerance mechanism makes a cloud self-subsistent to detect, identify and recover from interrupted state to normalized operation. Fault tolerance is also crucial for any system as it permits it to offer the intended services even in the presence of failure ranging from individual

ISSN: 0971-1260
Vol-22-Issue-17-September-2019

components to a bunch of jobs [7]. For an effective fault tolerance mechanism, we have to consider some parameters like -
a) System must be capable to adopt its environment without affecting its functionality usually it is termed as adaptability [7].

b) There must be some mechanism to keep count of tasks completed per unit time. It is being addressed as Throughput. Which gives no. of tasks completed successfully before the occurrence of the fault [7].

c) The count of additional resources required by the system to recover from the fault. We address it as scalability. It is needs to be on the lowest count of it [7].

d) It is on a positive side to have such a system which needs less time for responding to user requests. We address this part as response time. Ideally it should be on its lower value [7].

e) The count of resources required to by user to complete a specific task should be on its lowest value. We address this aspect as usability.

f) The no. of instances a resource is available to user at point of time is also important factor. It is addressed as availability. Ideally it should be on its higher value.

g) The usage of additional resources needed by the fault tolerance mechanism to recover the system from fault is addresses as overhead associated. Ideally it should be to its minimum level when fault tolerance mechanism is operational [7].

IV – Existing Fault Tolerance Techniques

Cloud computing is an open environment providing services to the users as per their needs. It also encounters no of faults while responding to the users for different request. Dealing with these resources manually will be difficult job. Count of fault tolerance techniques are existing which helps to resolve these faults. We can classify them on the basis how they respond to fault like –

Proactive techniques for fault tolerance clouds

These are the techniques which repeatedly monitor the system in order to detect the symptoms of existence of fault. Which will help out to prevent the effects of fault before it can occur. It usually keeps count of methods as –

a) The system is bound to restart periodically which takes the system to its clean state with each restart. Usually it is addressed as rejuvenation of the software [8].
b) Ability of the system to automatically detect, diagnose and repair software as well as hardware faults. This ability of the system is addressed as self-healing by the system during a fault [9].
c) System can be configured in such a way that it will be able to manage the components which are about to fail in such a way that it will not impact the overall performance for the system. It is done by isolating the faulty components from their actual nodes or by replacing the faulty components with new one dynamically [10].

ISSN: 0971-1260
Vol-22-Issue-17-September-2019

Reactive techniques for fault tolerance clouds

Rather then monitoring the system these techniques usually works on recovery of the system. The state of the system is continuously saved and used while recovery is under process. Some of the commonly used reactive techniques comprise of methods like –

a) The current state of the system is continuously saved in case of failure and system is restarted from the most recent state. It is highly recommended for the jobs with longer duration [2].
b) Component replication and deployment from multiple resources also helps in recovery of the system in case of failure of the system. As it ensures execution of job at any cost [11].
c) Workload can be distributed among the nodes that are forming the cloud while ensuring the nodes are not overwhelmed and therefore prevent a faulty state of the system [12].

Machine learning based resilience fault tolerance techniques

This technique is sharing the features of reactive and proactive techniques which makes it able to predict faults and take corrective actions in order to minimize or avoid effects of the fault on the over all performance of the system. But this technique is equipped with capability to incorporate intelligent learning by continuous communication with current state and work environment [13]. These techniques are deployed as –

a) Response of the system are recorded for some critical situation. If system get crashed, exception handling mechanisms are deployed along with improvement in resiliency of the system. This technique is also address as fault induction [14].
b) The systems are enabled with capability to learn from their current operational environment and manage their fault handling strategies accordingly. Here the reinforced learning approach of machine learning is making its mark to make the system subsistent [15]

V - Conclusions

In this paper we have discussed about possible faults in clouds and they can be categorized, characteristics of effective mechanism for making the cloud fault tolerant, existing fault tolerant techniques like proactive and reactive techniques. The main emphasize is needed to be put on the machine learning based solutions like resilience methods as it will be helping the system to be self-subsistent. The current cloud is moving towards there next level which will also increase the no. of faults as well as which might not possible to dealt with conventional fault tolerance approaches. So, it suggested here that we have to come up with systems which can learn and adapt the operational environments. These can happen when the systems will use machine learning methods as component of their fault tolerance strategies.

References

ISSN: 0971-1260
Vol-22-Issue-17-September-2019

[1] Cheraghlou, Mehdi Nazari, Ahmad Khadem-Zadeh, and Majid Haghparast. "A survey of fault tolerance architecture in cloud computing." *Journal of Network and Computer Applications* 61 (2016): 81-92.
[2] Jhawar, Ravi, and Vincenzo Piuri. "Fault tolerance and resilience in cloud computing environments." In Computer and information security handbook, pp. 165-181. Morgan Kaufmann, 2017.
[3] Vishwanath, Kashi Venkatesh, and Nachiappan Nagappan. "Characterizing cloud computing hardware reliability." In Proceedings of the 1st ACM symposium on Cloud computing, pp. 193-204. ACM, 2010.
[4] Gill, Phillipa, Navendu Jain, and Nachiappan Nagappan. "Understanding network failures in data centers: measurement, analysis, and implications." ACM SIGCOMM Computer Communication Review 41, no. 4 (2011): 350-361.
[5] Prathiba, Soma, and S. Sowvarnica. "Survey of failures and fault tolerance in cloud." In 2017 2nd International Conference on Computing and Communications Technologies (ICCCT), pp. 169-172. IEEE, 2017.
[6] Mukwevho, Mukosi Abraham, and Turgay Celik. "Toward a smart cloud: A review of fault-tolerance methods in cloud systems." IEEE Transactions on Services Computing (2018).
[7] Kumari P, Kaur P., "A Survey of Fault Tolerance in Cloud Computing," Journal of King Saud University - Computer and Information Sciences (2018)
[8] Hanmer, Robert. "Software rejuvenation." In Proceedings of the 17th Conference on Pattern Languages of Programs, p. 21. ACM, 2010.
[9] Angarita, Rafael, Marta Rukoz, Maude Manouvrier, and Yudith Cardinale. "A knowledge-based approach for self-healing service-oriented applications." In Proceedings of the 8th International Conference on Management of Digital EcoSystems, pp. 1-8. ACM, 2016.
[10] Nagarajan, Arun Babu, Frank Mueller, Christian Engelmann, and Stephen L. Scott. "Proactive fault tolerance for HPC with Xen virtualization." In Proceedings of the 21st annual international conference on Supercomputing, pp. 23-32. ACM, 2007.
[11] Balasubramanian, Bharath, and Vijay K. Garg. "Fault tolerance in distributed systems using fused data structures." IEEE transactions on parallel and distributed systems 24, no. 4 (2012): 701-715.
[12] F. Saffre, R. Tateson, J. Halloy, M. Shackleton, and J. L. Deneubourg, "Aggregation dynamics in overlay networks and their implications for self-organized distributed applications," The Computer Journal, vol. 52, no. 4, pp. 397–412, Jul 2009.
[13] F. Farivar and M. N. Ahmadabadi, "Continuous reinforcement learning to robust fault tolerant control for a class of unknown nonlinear systems," Applied Soft Computing, vol. 37, pp. 702–714, 2015.
[14] Y. Kwon, M. Balazinska, and A. Greenberg, "Faulttolerant stream processing using a distributed, replicated file system," Proceedings of the VLDB Endowment, vol. 1, no. 1, pp. 574–585, Aug 2008.
[15] C.-Z. Xu, J. Rao, and X. Bu, "Url: A unified reinforcement learning approach for autonomic cloud management,"Journal of Parallel and Distributed Computing, vol. 72, no. 2, pp. 95–105, 2012.