International Journal of Trend in Scientific Research and Development (IJTSRD)
International Open Access Journal | www.ijtsrd.com
ISSN No: 2456 - 6470 | Volume - 2 | Issue – 6 | Sep – Oct 2018

Fault Detection and Prediction in Cloud Computing

Swetha. S, Dr. S. Venkatesh kumar
Department of Computer Applications, Dr. Sns Rajalakshmi College of Arts & Science,
Coimbatore, Tamil Nadu, India

ABSTRACT
Cloud computing is a new technology in distributed computing. Usage of Cloud computing is increasing quickly day by day. In order to help the customers and businesses agreeably, fault occurring in datacenters and servers must be detected and predicted efficiently in order to launch mechanisms to bear the failures occurred. Failure in one of the hosted datacenters may broadcast to other datacenters and make the situation of poorer quality. In order to prevent such circumstances, one can predict a failure flourishing throughout the cloud computing system and launch mechanisms to deal with it proactively.

Keyword: *Cloud computing, failure detection, cloud datacenters, probability and statistics, Bayesian probability, machine learning Datacenters, failure detection, failure management, dependable computing, coordinated fault propagation, IPMI, FTB, and Clusters.*

INTRODUCTION
Cloud computing is a new technology in distributed computing. Usage of Cloud computing is increasing quickly day by day. In order to help the customers and businesses agreeably, fault occurring in datacenters and servers must be detected and predicted efficiently in order to launch mechanisms to bear the failures occurred. Failure in one of the hosted datacenters may broadcast to other datacenters and make the situation worse. In order to prevent such situations, one can predict a failure proliferating throughout the cloud computing system and launch mechanisms to deal with it proactively. One of the ways to predict failures is to train a machine to predict failure on the basis of e-mails or logs passed between various components of the cloud. In the training session, the machine can identify certain message patterns connecting to failure of datacenters. Later on, the machine can be used to check whether a certain group of e-mails, logs follow such patterns or not. Additionally, each cloud server can be defined by a state which indicates whether the cloud is running properly or it is facing some failure. Limitations such as CPU usage, memory usage etc. can be maintained for each of the servers.

LITERATURE SURVEY:
- Accessibility directly depends upon how fast the cloud structure can detect any errors and take necessary steps to troubleshoot the problem.
- It is a major test for service providers to provide stable service or else it may cause huge financial loss for organizations.

PROBLEM STATEMENT:
- The large scale and dynamic nature of cloud has added extra difficulty when it comes to fault detection and management.
- While it is true that effective fault detection and prediction is serious, one should also know the reasons that led to the fault.

Pre-process:
Firstly, we derive the message patterns from the recorded messages in a message logs.

Extracting the failure information:
As discussed above, messages usually include a field which signifies priority information and helps the administrators to handle the messages according to their severity.

Message pattern generation:
- A message pattern is defined as a set of message types in the message window.
- The message pattern can be expressed as aorder of messages by either considering or overlooking their order.

Extracting the failure information:
Messages generally include afield which signifies priority information and helps the administrators to handle the messages according to their severity.

FAILURE DETECTION AND PREDICTION MECHANISMS

- We may label the runtime health related data with one of two classes, Class 0 for normal behavior and Class 1 for situations with failures. Then, Class 1 is very unusual compared with Class 0.
- In addition, data from the unusual class may be incomplete because of some collection problems.

Ensemble of Bayesian Models for Failure Detection

- A data point is labeled as normal or failure based on its probability of appearance as a normal data point.
- To construct the probabilistic model and assure high detection precision, we develop an ensemble of Bayesian sub models to represent a multi model probability delivery.

Decision Tree for Failure Prediction

- The failure detection method based on an ensemble of Bayesian models presented in the preceding section identifies anomalous behaviors in a data center. The anomalies are reported to the system administrations for verification under failures.
- The goodness of a split is measured by impurity. A split is pure if after the split, for all branches, all the data taking a branch belong to the same class. We use entropy to quantify impurity.

BACKGROUND

Fault-Tolerance Backplane (FTB)

- The CIFTS Fault Tolerance Backplane is an asynchronous messaging backplane that provides communication among the various system software components. The Fault Tolerance Backplane (FTB) provides a common infrastructure for the Operating System, Libraries and Applications to exchange information related to hardware and software failures.
- Different components can subscribe tobe alerted about one or more events of interest from other components, as well as notify other components about the faults it detects. The FTB framework comprises a set of distributed daemons called FTB Agents which contain the bulk of the FTB logic and manage most of the event communication throughout the system.

Intelligent Platform Management Interface (IPMI)

- The Intelligent Platform Management Interface (IPMI) defines a set of common interfaces to a computer system which can be used to monitor system health.
- The BMC connects to SCs within the same chassis through the Intelligent Platform Management Bus/Bridge (IPMB). Among other pieces of information, IPMI maintains a Sensor Data Records (SDR) repository which provides the readings.

DESIGN AND IMPLEMENTATION

- FTB-IPMI is designed to run as a single stand-alone muse which handles multiple operations like reading IPMI sensors, classifying events based on severity, and propagating the fault information via FTB.
- A single instance of the FTB-IPMI muse running on one node can manage an entire cluster. Once adjusted, the following actions are performed at periodic user-set intervals.

CLOUD USAGE

- Private Clouds are always owned by the respective enterprises. Functionalities are not directly visible to the customer, though in some cases services with cloud enhanced features may be offered this is similar to Software as a Service.
  - Example: eBay.
- Public Clouds enterprises may use cloud functionality from others, respectively offer their own services to users outside of the company.

TYPES OF FAULTS

These faults can be classified on several factors such as:

Network fault: A Fault occur in a network due to network partition, 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, etc.

Media faults: Fault occurs due to media head crashes.

Processor faults: fault occurs in the processor due to operating system crashes.

Process faults: A fault which occurs due to shortage of resource, software bugs, etc.

Service expiry fault: The service time of a resource will expire when some applications used by it.

A failure occurs during computation on system resources can be classified as:

- OMISSION FAILURE
- TIMING FAILURE
- RESPONSE FAILURE
- CRASH FAILURE

Permanent: These failures occur by accidentally by a wire cut, power breakdowns and etc. It is easy to reproduce these failures. These failures can cause major disruptions and some part of the system may not be functioning as desired.

Intermittent: These are some of the failures that appear occasionally. Mostly these failures are ignored while testing the system and only appear when the system goes into operation. Therefore, it is hard to predict the extent of damage these failures can bring to the system.

Transient: These are some failures that are caused by some inherent fault in the system. As these failures are corrected by retrying roll back the system to previous state such as restarting software or resending a message.

CONCLUSION:

Despite the many advantages offered by cloud computing, there are also networking concerns that creel its fast implementation. This article has reviewed and analyzed the networking-related issues that arise due to resource outsourcing, the virtualized, shared, and public nature of cloud computing, the emerging challenges from security breaches, and the increasing need to provide a resilient cloud computing infrastructure and services. This discussion also presented and examined related contributions from industry, academia and correction fields. Finally, the article also highlighted relevant cloud computing areas requiring further research.

REFERENCE

1. Agarwal, Sharad, John Dunagan, Navendu Jain, Stefan Saroiu, Alec Wolman, and Harbinder Bhogan. 2010. Volley: Automated Data Placement for Geo-distributed Cloud Services. Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation. Berkeley, CA, USA: USENIX Association, 16 pages.
2. Alamri, Atif, Wasai Shadab Ansari, Mohammad Mehedi Hassan, M. Shamim Hossain, Abdulhameed Alelaiwi, and M. Anwar Hossain. 2013. A Survey on Sensor-Cloud: Architecture, Applications, and Approaches. International Journal of Distributed Sensor Networks, 18 pages.
3. Ali, M., Khan, S., Vasilakos, A.. 2015. Security in cloud computing: Opportunities and challenges, Information Sciences, 305(1), 357-383.
4. Buyya, Rajkumar, et al. "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility." Future Generation computer systems 25.6 (2009): 599-616.
5. Zhang, Qi, Lu Cheng, and Raouf Boutaba. "Cloud computing: state-of-the-art and research challenges." Journal of internet services and applications 1.1 (2010): 7-18.
6. Y. Watanabe, H. Otsuka, M. Sonoda, S. Kikuchi and Y. Matsumoto, "Online failure prediction in cloud datacenters by real- time message pattern learning," 4th IEEE International Conference on Cloud Computing Technology and Science Proceedings, Taipei, 2012, pp. 504-511. doi: 10.1109/CloudCom.2012.6427566
7. Guan, Qiang, Ziming Zhang, and Song Fu. "Ensemble of bayesian predictors and decision trees for proactive failure management in cloud computing systems." Journal of Communications 7.1 (2012): 52-61.
8. Guan, Qiang, Ziming Zhang, and Song Fu. "Proactive failure management by integrated unsupervised and semi-supervised learning for dependable cloud systems." Availability, Reliability and Security (ARES), 2011 Sixth International Conference on. IEEE,
9. R. K. Sahoo, A. J. Oliner, I. Rish, M. Gupta, J. E. Moreira, S. Ma, R. Vilalta, and A. Sivasubramaniam, “Critical event prediction for proactive management in large-scale computer clusters,” In Proceedings of ACM International Conference on Knowledge Discovery and Data Dining (KDD), 2003.
10. A. J. Oliner, R. K. Sahoo, J. E. Moreira, M. Gupta, and A. Sivasubramaniam, “Fault-aware job scheduling for BlueGene/Lsystems,” In Proceedings of IEEE/ACM International Parallel and Distributed Processing Symposium (IPDPS), 2004