SELF DIAGNOSIS AND SELF HEALING TECHNOLOGY IN CLOUD COMPUTING

MENAKA C
Asst. Professor,
Research Scholar (Bharathiaar University),
Assistant Professor (St. Claret College)
Bangalore

PRAVEEN KUMAR V
Asst. Professor,
Acharya Bangalore B School,
Bangalore University,
Bangalore

Abstract

Cloud computing requires a robust, scalable and high performance infrastructure. To provide a reliable and dependable cloud computing platform, it is necessary to build a self-diagnosis and self-healing system against various failures or downgrades. This paper describes the self-healing function, a challenging topic in today's cloud computing systems, from the consequence – oriented point of views and the usefulness of autonomic computing in cloud computing.

Keywords: Self-Healing, Consequence Oriented, Autonomic computing and Cloud Computing

I. INTRODUCTION

An evolutionary trend in computing has started to take place called cloud computing, which aims to realize massive – scale service sharing over the internet. Cloud computing is a general term for anything that involves delivering hosted services over the Internet. These services are broadly divided into four categories: Infrastructure – as – a Service (IaaS), Platform – as – a Service (PaaS), Software – as – a Service (SaaS) and Data – as –a Service (DaaS). Cloud computing service providers will need a robust, scalable and high performance infrastructure to compete in this emerging market, therefore, the many challenges involved in adopting cloud computing must be thoroughly investigated.

A cloud can be defined as a pool of computer resources that can host a variety of different workloads, including batch – style back end jobs and interactive user applications [1]. Clouds are used by a huge number of people.

As computer systems become increasingly large and complex, their dependability and autonomy play a critical role in supporting next – generation science, engineering and commercial applications [2 – 3]. Clouds offer the automatic resizing of virtualized resources. Cloud scalability is mainly enabled by increasing the number of working nodes and requires dynamic self – configuration in accordance with high level policies. Cloud computing provides computing services to large pools of users and applications, and thus, clouds are exposed to a number of dangers such as accidental /deliberate faults, virus infections, system failures etc., [4]

As a result, clouds often fail, become compromised or perform poorly, therefore becoming unreliable. Consequently in cloud computing, the challenge to design, analyze, evaluate and improve dependability remains. Self-diagnosis and self-healing is one of the important ability with endowing clouds to reduce the complexity and increase the sustainability to the failures.

A possible solution may reside in autonomic computing that enables system to manage them without direct human intervention.Automatic computing is about shifting the burden of managing systems from people to technologies. The aim of autonomic computing is to develop the systems' capabilities of self – managing the distributed computing resources to adapt to unpredictable changes while hiding the system's intrinsic complexity to users. The idea of autonomic computing is to mimic the autonomic nervous systems found in biology, as first presented by IBM researchers [5] and extensively studied in follow up research .[6 -8]

Autonomic computing refers to the self-managing characteristics of distributed computing resources, adapting to unpredictable changes whilst hiding intrinsic complexity to operators and users. As name implies an autonomic system makes decisions on its own, using high-level policies such as service level agreement; it will constantly check and optimize its status and automatically adapt itself to changing conditions.

It does not have any Artificial Intelligent (AI) involve with it, it is just control theory. Having said that Autonomic computing can be modeled in terms of two main control loops (local and global) with sensors (for self-monitoring), effectors (for self-adjustment), knowledge and planer/adapter for exploiting policies based on self- and environment awareness.

Typical Autonomic Computing system will contains following characteristics,

- Self-Configuration: Automatic configuration of components.
- Self-Healing: Automatic discovery, and correction of faults.
- Self-Optimization: Automatic monitoring and control of resources to ensure the optimal functioning with respect to the defined requirements.
- Self-Protection: Proactive identification and protection from arbitrary attacks.[9]

To provide a reliable and dependable cloud computing platform, it is necessary to build a self-diagnosis and self-healing system against various failures or downgrades.

II.Self-diagnosis and self-healing

Self-healing, automatically detecting errors and recovering from failures is a challenging topic in cloud computing. Current self-healing is far from perfect. Most self – healing systems diagnose and heal the failure after failure occurs rather than anticipating failures. Accurately self-healing instead needs to involve more complicated computation, analysis and decision processes, analyzing system level models and making decisions in a holistic manner. Suspect events including abnormal cases that are detected should lead to the analysis to anticipate / fore stall failure.

Traditional mechanisms for diagnosis and healing test and locate bugs that should be removed from software codes. However, such procedures have to stop the program and recompile it, which is not suitable for the self-healing function in cloud computing because the hosts in cloud are massive and are not expected to reboot or stop programs for recompilation. Moreover, the bugs and locations of the problems can vary, so it is impractical to ask the host itself to precisely locate errors and to intelligently remove them from the codes as if a professional programmer.

A software system may contain various faults / errors that reside in different modules, classes or lines and lead to some symptoms. These symptoms can be monitored in the real time system. The influences of the bugs, the symptoms in the system, are accumulated so as to cause a certain consequences that affect the overall performance and quality of the service or task. Gleaning the symptoms through monitoring the host predicts or diagnosis the possible consequences from the symptoms. [10 -11]

The consequence oriented concept is predictive and preventative by diagnosing the symptoms detected in the real time, before the catastrophic failures really occur. This new concept can prevent serious consequence derived from the monitored symptoms in advance.

III. Consequence – Oriented self – diagnosis

Consequence oriented diagnosis diagnoses the severity levels, according to the symptoms. Some diagnosis and prescriptions are time consuming and can only be applied in situations with major symptoms that are not very serious; they cannot be applied to serious problems which may quickly bring the system down. At this point, some quick diagnoses and prescriptions should be applied in order to prevent complete failure as obtaining a precise diagnosis after the system has gone down or the mission has failed is worthless.

Therefore, the severity levels of the consequences should be diagnosed first. There are several requirements for the diagnosis of consequence severity. First it should be quick and straight forward because the severity level will determine the next step of consequence diagnosis and start the different categories of the healing method. Second, the definition of the severity levels should be related to the degree, the system can afford the detected symptoms because the timing for healing is critical. Third, the severity levels should also reflect how far the current state is away from a serious consequence, which tells an approximate duration for the next diagnosis and healing to prevent the occurrences of system failures.

After determining the severity levels, the consequence diagnosis enters the next phase. This phase first determines the category of consequence corresponding to the severity levels. Then, which diagnosis system to be used is decided based on the category because each category of consequence has its own diagnosis system trained particularly for a set of consequence.

IV. Consequence Oriented Self-Healing

Self-Healing needs to be implemented via three necessary components: Instructions, Healing Categories and Recursive Healing method.

As an important component in consequence oriented healing, we hereby propose the idea of instructions. The instruction ought to recover the affected host and processes from or prevent serious consequences or failures. However, different prescriptions are aimed at fulfilling different requirements, so it is possible for a certain prescription to satisfy some requirements but sacrifice others especially when some requirements are in conflicts or present tradeoffs.

The host will choose corresponding categories of self-healing approaches according to determined severity levels. Four typical healing categories are described here:

Minor severity level: The affected process is allowed to continue running. In the meantime, follow up diagnosis steps are executed to find the possible consequences and then decide the corresponding prescriptions to heal it.

Intermediate severity level: The affected process is suspended immediately and then the follow up diagnosis starts to detect the consequences and recover from or prevent the consequence.

Major severity level: If the affected process has the potential to cause the destruction of the entire system, then the fastest and simplest way is to kill this dangerous process. However, before the process becomes serious, the host should have initiated another backup process to run for the same function, though ineffectively.

Catastrophic level: If the host is detected with catastrophic failure, then reboot the system. After rebooting, the previously monitored and stored states of all processes can help restart the system from the latest checkpoint.[14]

More categories can be added as needed in reality.

V. Recursive Healing method:

After the diagnosis, the host determines an instruction to heal the identified problem.The first option is the host runs the instruction. The results after the healing are fed back. However, if the problem still persists, the second option the hybrid diagnosis can be tried to heal the problem or further diagnosis is completed and another instruction can be tried. This step can be recursively attempted for several rounds until one instruction works to alleviate the consequence. In the case all attempts are useless; the host can restart the corresponding processes or reboot the operating system. Before restarting the process, some important variables or states can be backed up in order to allow the system to recover back to the latest Check point. The healing results will also be reported to the diagnosis modules for learning. As it is possible that the affected host lacks sufficient resources to support the healing, in such cases, the system can assign another host to heal the affected host.

The above recursive healing method not only tries multiple instructions according to the possibilities assigned by the hybrid diagnosis but also utilizes other hosts to realize peer healing if the current host itself has inadequate resources to execute the instructions.

To implement the tool of self-diagnosis and self-healing, we have to integrate the following analysis method

VI. Gaussian influence Diagram

An influence diagram is a graphical representation for a decision problem under uncertainty. It has been used primarily to help decision makers and analyst structure models and communicate concepts of conditional independence and information flow. In this way assessment, evaluation and sensitivity analysis can all be performed on the model in the most natural form to the decision maker.

An influence diagram is a network consisting of a directed graph with no directed cycles. Each node in the graph represents a variable in the model. This variable can be either a constant, an uncertain quantity, a decision to be made or an objective.[13]

VII. Naïve Bayes Classifier

A NBC is a simple probabilistic classifier based on applying Bayes theorem. The Naives Bayes model is a specific form of Bayesian network which is widely used for classification. It combines the Naives Bayes probability model with a decision role. It is applied for consequence oriented diagnosis.

NBC is one of the most efficient and effective inductive learning algorithms for machine learning and data mining. The condition independent assumption is clearly violated in real world problems. However NBC often work better in many complex real world situations.[14]

VIII.Hybrid Tool

The hybrid tool is a diagnosis approach that represents a combination of the analytical methods co –operating together to realize the diagnosis purpose as shown in fig.1.

GID (Gaussian Influence Diagram) is integrated with NBC (Naives Bayes Classifier) to form a hybrid tool, In this model GID is applied to determine the severity levels, then the NBC is applied to infer the possible consequences. First, the severity level A output from the GID module for selecting the corresponding healing categories. The severity level A also feeds forward to NBC to know the consequences. The symptoms (C) are preprocessed to B and then feed into the NBC to derive the possibilities associated with corresponding consequences. Finally the results are the consequences D out of the diagnosis for self-healing. Then, the healing module selects and runs the corresponding instruction with the result E. After that NBC adjust the parameter according to the result.

IX. Conclusion

To provide a reliable and dependable cloud computing platform, it is difficult but obviously interesting to build a self-diagnosis and self-healing system to prevent / forestall various failures or downgrades. This paper describes and proposed a solution in self-healing that is based on the consequence oriented diagnosis and healing. Consequence oriented diagnosis offers the prediction of the consequences from the symptoms instead of detecting the causes that lead to the symptoms. More research should be conducted to pioneer future work in this area.

Reference

1. Erdogmus.H., Cloud Computing, IEEE Software, 26(2), 4-6-2009

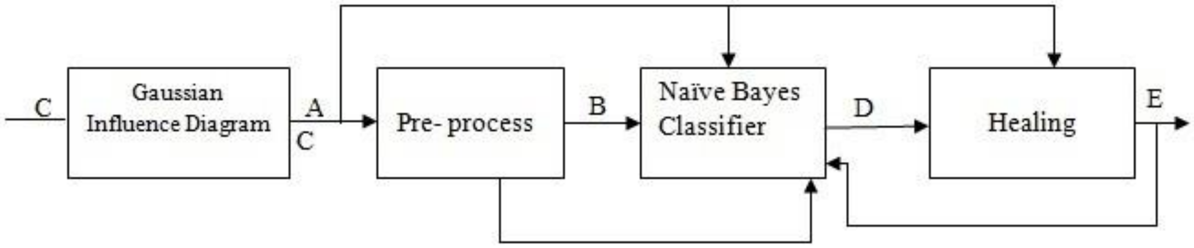

Fig 1: A Hybrid tool integrating Gaussian influence diagram with Naive Bayes Classifier

2. Dai, Y.S., Xie,M., Poh K.L., Markov Renewal models for correlated softwares failures of multiple types, IEEE Trans. Relia.

3. 54, 100 – 106 (2005)Dai. Y.S., Marshall.T., Guan, X.H., Autonomic and Dependable Computing; Moving Towards a modal-Driven Approach. J., Comput. Sci. 216, 496 – 504 (2006)

4. Xie M., Dai Y.S., Poh, K.L., Computing Systems Reliability Models and Analysis, kluwer Academic Publishers, New York (2004)

5. Kephart, J.O., Chess, D.M., : The Vision of Autonomic Computing, IEEE Computer 36(1), 41 -50 (2003)

6. Motuzenko P.: Adaptive Domain Model: Dealing with Multiple Attributes of Self managing Distributed object systems. In: 1st International Symposium on information and communication Technologies. PP. 549 – 554 (2003)

7. Paulson.L: Computer System, Heal Tryself. Computer 35(8), 20 – 22 (2002)

8. Patterson, D., Brown, A., Broadwell,P., et al.; Recovery oriented computing roc: Motivatin, Definition, Techniques and case studies Technical Report CSD -02-1175, Univ. of California – Berkeley (2002)

9. DeepalJayasinghe; http://blogs.deepal.ag/2009/01/autonomiccomputing.html

10. Hinchey, M., Dai, Y.S., Rash, J.L., Truszkowski, W., Madhusoodan M. Bionic Autonomic Nervous System and self-healing for NASA Ants Missions. In. 22nd Annual ACM Symposium on Applied Computing. PP. 90 - 96 (2007)

11. Dai,Y.S., Hinchey, M., Madhusoodan,M., Rash, J.L., Zou, X,: A Prototype model for self-healing and self-reproduction in swarm robotics system. In: 2nd IEEE symposium on Dependable, Autonomic and secure computing. PP 3 -11 (2006)

12. Ross D.S. Hachter& C. Robert kenley: Gaussian Influence Diagram, Management Science Vol. 35, No. 5, May 1989.

13. Domingus, P., Pazzani M., On the optimally of the simple Bayesian Classifier under zero – one Loss. Machine Learning 29. 103 – 137 (1997).

14. Yuanshun Dai, Yamping Xiang and GeweiZheng, Self-healing and hybrid diagnosis, Cloudcom 2009 LNCS 5931, PP 45 -56, 2009

****

******