Chapter 59
Efficient Fault Tolerance on Cloud Environments

Sam Goundar
CENTRUM Catolica Graduate School of Business, Pontificia Universidad del Peru, Peru

Akashdeep Bhardwaj
University of Petroleum & Energy Studies, Dehradun, India

ABSTRACT

With mission critical web applications and resources being hosted on cloud environments, and cloud services growing fast, the need for having greater level of service assurance regarding fault tolerance for availability and reliability has increased. The high priority now is ensuring a fault tolerant environment that can keep the systems up and running. To minimize the impact of downtime or accessibility failure due to systems, network devices or hardware, the expectations are that such failures need to be anticipated and handled proactively in fast, intelligent way. This article discusses the fault tolerance system for cloud computing environments, analyzes whether this is effective for Cloud environments.

1. INTRODUCTION

The growth of internet and cloud computing has transformed business opportunities globally. The availability of computing resources and IT services have risen form a low 90% to 99.999% for both corporate and non-business users. As more and more virtual business applications are being delivered over the internet to end users and corporate enterprise employees, cloud computing environment is evolving to deliver efficient services by innovative cloud models, multiple high availability devices, virtualized systems (Vishwanath et al., 2010). These also include multiple layers of abstraction, which turn the applications and infrastructure be more distributed and complex than ever before. On the other hand, end users have come to expect high level of fault tolerance and availability with swift and flawless execution of the hosted applications. Cloud providers and data center infrastructure management teams constantly strive to maintain this high level of availability and fault tolerance. Some of these methods are use of Application Performance Monitoring (Armbrust et al., 2010), having multiple devices connected in high

DOI: 10.4018/978-1-7998-5339-8.ch059

availability (HA) mode by over provisioning devices, having a hot swap Disaster Recovery (DR) site or Network Monitoring system in order to provide better fault tolerance in case of any downtime. Users expect their computing systems (Anju et al., 2012) have the ability to handle gracefully any unexpected system or application programming malfunction and provide seamless availability which in IT jargon is termed as fault tolerance as described below.

Fault Tolerance means that the loss of a service (the network itself, some host, or some critical software running on a host) is tolerated by the system (Wenbing et al., 2012). Usually, it means that there are enough other instances of that service available that the system can carry on using those other resources without significant impact to the system's responsiveness overall.

Load balancing means that a large workload is shared among many instances of a service (or many hosts, or even many instances of the service on many hosts) but doesn't guarantee fault tolerance, though it can help (Chen et al., 2010). If one of the available participants in the load balanced cluster fail, odds are there are enough resources available to continue satisfying requests. However, if the load balancer itself fails, the cluster might become useless. The load balancer itself might need to be fault tolerant - there might need to be two load balancers.

High availability ensures that a resource is available, even as the resource may suffer from some amount of minor downtime, Fault Tolerance (FT) can be defined as not losing (Kumar et al., 2011) that in-memory session state in event of a failure like having a host server crash or a network device failure rather than the service failing completely.

2. FAULT TOLERANCE FOR CLOUD ENVIRONMENTS

Fault Tolerance aims to ensure systems are able to deliver in case of one of more failures of the unit's components. Fault Tolerance (Anjali et al., 2016) is system resource availability and reliability not being affected in case any of the preceding component or execution devices (Mohammed et al., 2016) failing or there are multiple failures for the hosted application system or infrastructure devices (Zhang et al., 2011). Usually systems, devices or resources are often over provisioned or purposely underutilized to ensure even if the application performance might be affected during an outage, the systems continue to perform possibly at a reduced level, rather than failing completely within predictable and acceptable bounds. Fault tolerance is mostly implemented in high-availability life-critical system environments. Providing fault tolerant design (Patra et al., 2013) for each and every single component is however not an effective solution. The associated redundancy and over provisioning brings a number of parasitic penalties: increase in weight, cost, power, size, consumption, as well as time to design, verify and test before delivering the service. The following options are taken into account when determining how and why the computing components should be fault tolerant:

- **How Critical Is That Component?** In a data center, having a spare catalyst running idle is good to have but not critical, with low failure rate Catalyst switch would be low on fault tolerance while an extra Supervisor management module would be great to have.
- **How Likely Is the Component Expected to Fail?** Some components, like disk drives in SAN or Power supply in servers a car, are likely to fail, so fault tolerance is needed.