International Journal of Education and Science Research Review
E-ISSN 2348-6457
Volume-4, Issue-2 April - 2017 P-ISSN 2349-1817
www.ijesrr.org Email- editor@ijesrr.org

RESOURCE MANAGEMENT OF SERVICES, NETWORK AND ENVIRONMENT IN A UTILITY CLOUD

Vikas Jain
Assistant Professor
Dept. of Computer Application
C.C.S. University, Meerut

ABSTRACT
Cloud Computing customers are looking for the best utility for their money. Research shows that functional aspects are considered more important than service prices in customer buying decisions. Choosing the best service provider might be complicated since each provider may sell three kinds of services organized in three layers: SaaS (Software as a service), PaaS (Platform as a service) and IaaS (Infrastructure as a service). In this paper, Cloud computing is known to lower costs of corporate Information Technology (IT). Thus enterprises are eager to move IT applications into public or private cloud. Because of this trend, networks connecting enterprises and cloud providers now play a critical role in delivering high quality cloud applications. Simply buying better devices is not viable for improving network quality, due to high capital costs. A more attractive approach is to better utilize network resources with proper network management. However, there are two problems with current network management: separately managing network components along the end-to-end path, and heavily relying on vendor-specific interfaces with devices.
KEYWORDS: Cloud computing, network resource

INTRODUCTION

Cloud computing is reshaping the Information Technology (IT) industry. It offers utility computing by delivering the applications as services over the Internet and providing these services with well-organized hardware and software in datacenters. The utility computing model eliminates the large capital barrier of purchasing hardware for enterprises, and it also lowers the IT operational costs by allowing enterprises to pay for what they actually use. These benefits motivate an ongoing effort of enterprises to move their IT applications into the public cloud and/or build their private cloud. The delivery of the promise of cloud computing depends on the quality of the end-to-end network. As shown in Figure 1, the Internet now plays a critical role in carrying the traffic of IT applications between enterprises and datacenters of public/private cloud providers. The performance of cloud-based IT applications depends on not only the application software in the cloud, but also the reliability, efficiency, and performance of the networks in the middle. Improving the network quality could be achieved by deploying more network devices with higher bandwidth. However, this brute-force approach no longer works due to the high capital costs and the rapidly growing traffic demands. A more attractive approach is to build proper network management solutions to better utilize the existing network resources.

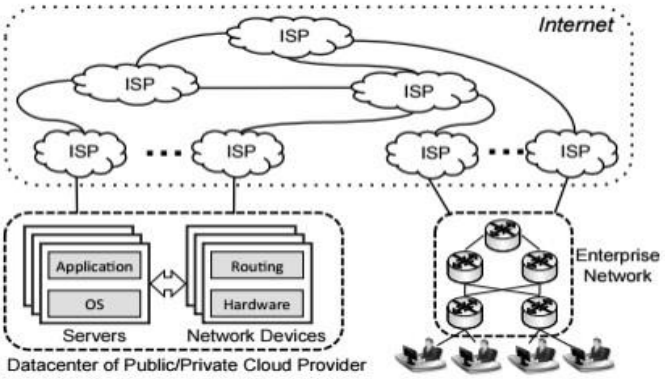

Fig. 1 End-to-end Structure of Cloud-based Software Services

DISJOINT MANAGEMENT OF THE END-TO-END COMPONENTS

Providing an efficient network involves multiple components on the end-to-end path: from the network stack of the datacenter servers' operating systems, the hardware configuration and the routing control of network devices in datacenters, the traffic exchange of Internet service providers (ISPs), to the setup of enterprise networks. Each component can affect the quality of the networks. Yet these components (e.g., servers, network device hardware, and traffic routing) are managed separately with different systems, since traditionally these components are spread across multiple places. In the cloud era, these components become much more concentrated in the datacenters than before, and the lack of integration among management systems limits the quality improvement of the networks.

LOW-LEVEL INTERFACES FOR INTERACTING WITH NETWORK DEVICES

Network devices are heterogeneous with different models from various vendors of varying ages. Interacting with the devices is complicated since the configurations APIs with devices are usually low-level and vendor-specific. Human network operators have to heavily use these low-level APIs in day-to-day operations, and it has been an error-prone process to configure the devices to run the right protocols with the right parameters using the right APIs. Using these APIs also tightly binds the solutions to specific device features by vendors, making it difficult to adapt the solutions to evolving business objectives. In datacenters, the heavy reliance on low-level device interfaces becomes one of the major sources of failures in network operations, especially when the datacenter network is growing in scale and adopting more commodity hardware from multiple vendors.

The programmability of management solutions has received much attention in the research community. The concept of Software Defined Networking (SDN) aims at providing a better way to program traffic management solutions. The literature on SDN, especially the ones surrounding the Open Flow technology, promise to automate managing the traffic routing in networks with high-level programming paradigms. With existing literature focusing on programming traffic management on network devices, two other problems are much less explored beyond just traffic management: disjoint management of network components (e.g., server, device hardware), and low-level device interaction limiting a broader scope of network management (e.g., infrastructure management).

END-HOST/NETWORK COOPERATIVE TRAFFIC MANAGEMENT

The providers of cloud services and cloud-based applications carefully manage their traffic through the underlying networks for various performance objectives. These traffic management solutions are confined in the scope of network devices. In addition to the limited CPU and memory resources, network devices cannot provide knowledge in layers higher than the network layer, limiting the solutions' insights into the application-traffic behaviors. We propose to join the end hosts with the network devices, so we can utilize the rich application-traffic statistics in the end hosts to build better traffic management solutions. Our system, named Hone, abstracts the diverse collection of statistics on both end hosts and network devices into a uniform view of data. We then design a framework based on functional reactive programming to simplify programming management solutions with the uniform data model. Hone has been adopted by Overture Networks to use in the Verizon Business Cloud service. With the host side data from Hone, the customers of Verizion Business Cloud enjoy better quality of their connections with the datacenters of Verizon for the improved performance of cloud-based applications. Research work on Hone is published in Springer Journal of Network and Systems Management.

HONE FOR MULTI-TENANT CLOUD ENVIRONMENT

Hone collects the fine-grained traffic statistics from inside the end hosts, assuming that the cloud providers have access to the hosts' operating systems. In a multitenant public cloud, tenants may not want the cloud providers to access the guest OS of the virtual machines. A viable alternative would be to collect measurement data from the hypervisor and infer the transport-layer statistics of the applications in the virtual

machines. This direction is currently under exploration, and can complement Hone to support more types of cloud environments.

REVIEW OF LITERATURE

In this section an in depth study of three previous works done in this area is briefly discussed with respect to their limitations as far as their proposed frameworks. Firstly, Hwee (n.d.) proposed a framework generalizing his findings found as results of studying only one secondary school in the whole of Singapore and generalize the finding. The sample size is not enough representation of all the numerous schools in Singapore. The author identified key factors of trust, comfort level, command of language, attitude towards work and image, as additional factors that would influence the adoption of Social Networks.

Secondly, Venkatesh et al. (2003) combined eight technology acceptance models to propose a framework known as the Unified Theory of Acceptance and Use of Technology (UTAUT). This model encompasses the following factors performance expectancy, expectancy effort, social influence, facilitating conditions and behavioural intention. In addition to these factors other moderators (gender, age, voluntaries, and experience) were used to measure the influences on the factors. Though extensive work was done in this area the following setbacks were identified: lacks other key contextual factors when it comes to the use and implementation of social networks. Indicators such as policy and culture and financial support were not discussed in the work of Venkatesh et al. (2003) and the authors were silent on the acquisition and implementation factors.

Finally, the work presented by Monguatosha et al.(2016) discusses two critical factors in addition to the existing factors on the UTAUT model. In their work, budgeting and accountability (BA) and organizational culture (OC) was added as additional factors that could influence the use of SN's adding up to facilitation conditions. The Technology Acceptance Model (TAM) was used in the proposition made by Monguatosha et al. (2014).

MOTIVATION

Our research was motivated by a practical scenario at our university's data centre. In the (not so distant) past, we applied the "traditional architecture" which was composed of diverse processing clusters configured to process different services. We faced the usual issues encountered in large data centres at that time: lack of rack space, which impacted flexibility and scalability; an excessive number of (usually outdated) servers, which impacted operation costs; the need of an expensive refrigeration system; and an ineffective Uninterruptible Power Supply (UPS) system, which was problematic to scale due to the number of servers involved. With the use of cloud computing, we managed to consolidate the number of servers using virtualisation techniques. Using this technology, we concentrated the predicted load on a few machines and kept the other servers on standby to take care of peak loads. The immediate results were very positive: reduction of rack space utilisation; lower heat emission due to the reduction in server utilisation, with consequent optimisation of the cooling infrastructure, and, a quick fix for the problematic UPS system because we had less active servers. As part of an institutional initiative towards sustainability and eco-friendliness, our next step was to optimise energy utilisation and reduce carbon emission. For this, we looked at solutions from the fields of computing and, more specifically, cloud computing. We noticed that there was room for improvement as we consolidated resources using cloud computing. For instance, there were periods in time when the Virtual Machines (VM) were idle and the servers were underutilized. Based on the principles established by Buyya et al. (2017), our goal was to promote energy-efficient management and search for methods to safely turn off unused servers using an on-demand basis. The intuitive approach was to concentrate the running applications (configured per VMs) in a few servers and recycle server capacity. Although appealing, this approach led to a major issue: service unavailability! A quick analysis concluded that it was related to the time required to bring up the servers during unpredictable peak loads. We concluded

the following: (i) the dimensioning is based on historic intra-day analysis of services demand. More specifically, it is based on the analysis of previous day's demand plus a margin of the business growth that can be estimated as the amount of resources required for one service in a period of time; (ii) however, when dealing with services with highly variable workloads, that prediction becomes complex and often immature. Moreover, external factors can lead to unexpected peaks of demand. For that, we left a safety margin of resources available (e.g. 20% extra resources on standby). Besides the excessive energy utilization, this approach fails when the demand surpassed that threshold; (iii) as a solution, we needed to bring up turned-off resources. The lapse of time between the detection of the situation and the moment that processing resources become available caused the service unavailability. We analysed several alternatives to overcome this issue that implements an organisation theory model for integrated management of the clouds focusing on: (i) optimising resource allocation through predictive models; (ii) coordinating control over the multiple elements, reducing the infrastructure utilization; (iii) promoting the balance between local and remote resources; and (iv) aggregating energy management of network devices. Our decision was in favour of these options as it addresses the core problem that is inefficient management, reliability, and cost reduction.

FRAMEWORK FOR SOCIAL NETWORK SERVICE SELECTION

The social network organizations are clustered in groups based on their objectives. Each group (G) consists of a set of social network organizations. In general

G n = {O n1 , O n 2 ,..., O nm }

Where n is the number of group and m is the number of services provided by a single organization. The social network organizations and Users that are available in a particular group are handled by Social Network Service Authority SNSA1. The communication between Social Network Service Authority's takes place through Master Social Network Service Authority MSNSA. The Fig. 2 show the proposed framework for social network service selection. The Social Network Service Authority consists of Service Registration Unit, User Registration Unit, Social Network Service (SNS) Selection Unit and Social Network Service (SNS) Delivery Unit. Each social network organization furnishes the services it wishes to provide and registers it with the service registration unit.

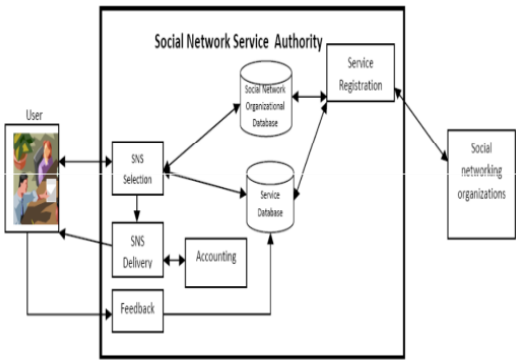

Fig. 2 Proposed Framework for Social Network Service Selection

CONCLUSION

An effective framework for selection of social network services was proposed. Many criteria such as information richness, reliability, search ability, paid/unpaid services, customizability, platform independence and activity support that influence the service selection were considered. The experimental analysis was performed on the proposed framework and was found to be effective.

REFERENCES

1. Alejandro Zunino, Marcelo Campo, "Easy web service discovery: A query-by-example approach", Science of Computer Programming, vol.71, pp.144–164, 2018
2. Anandha Gopalan, Taieb Znati, "SARA- A service architecture for resource aware ubiquitous Environments", Pervasive and Mobile Computing, vol. 6, no.1, pp. 1-20, 2014.
3. Dimitrios Zissis, Dimitrios Lekkas, "Addressing cloud computing security issues", Future Generation Computer Systems, vol.28, no.3, pp.583 – 592, 2012.
4. El Khati H., Bash R., "A framework and QoS Matchmaking Algorithm for Dynamic Web Services Selection", The Second International Conference on Innovations in Information Technology, Dubai, UAE, 2015.
5. Kirubakaran Ezra, Elijah Blessing Rajsingh, "Scalable and Reliable Methodology for Service Selection in Pervasive Computing", Proceedings of 3rd IEEE International Conference on Electronics Computer Technology, vol.1, pp. 188-191, ISBN: 978-1-4244-8678-6, 2017.
6. Nicola Marsden, Peter Kübler, Corinna Leonhardt, Sabine Thomanek, Hartmut Jung, Annette Becker, "Specifying computerbased counseling systems in health care: A new approach to user-interface and interaction design", Journal of Biomedical Informatics, vol.42, pp.347–355, 2009.
7. Won-Sik Yoon, "Ubiquitous Service Discovery in Pervasive Computing Environment", Information Technology Journal, vol.7, pp.533-536, 2015.
8. Yuping Yang, Nick Taylor, Sarah McBurney, Elizabeth Papadopoulou, Fiona Mahon, Micheal Crotty, " Personalized Dynamic Composition of Services and Resources in a Wireless Pervasive Computing Environment", Proceedings of 1st International Symposium on Wireless Pervasive Computing, pp. 1-6, 2016.