Advanced Science and Technology Letters
Vol.120 (CCSC 2015), pp.742-746
http://dx.doi.org/10.14257/astl.2015.120.147

Virtualized Testbed Development using Openstack

Byeongok Kwak[1], Heeyoung Jung[1],

[1] Electronics and Telecommunications Research Institute (ETRI),
218 Gajeong-Ro, Yuseong-Gu, Daejeon, S. Korea
{kwakbok, hyjung}@etri.re.kr

Abstract. The cloud computing paradigm promises to deliver hardware to the end user at a low cost with an easy to use interface via the internet. Due to the rapid development of open source based Cloud management platforms such as OpenStack, many and various individuals and institutions are trying to construct Cloud computing systems based on open source software projects. This paper was tested by constructing a virtualized testbed environment using openstack as IaaS to test the MOFI function. We were using the KVM as a hypervisor for virtualization system construction and the OpenStack for the virtual machine VM management. In the operating system, we were using Ubuntu 14.04. Virtualized testbed environment, in order to ensure interlocking and reliability with the physical test environment was configured in conjunction with the actual network.

Keywords: Virtualized testbed, cloud computing, openstack

1 Introduction

The term cloud computing is often applied to diverse array of internet based computing services, and providing a concrete definition is often difficult. In a broad sense, cloud computing is designed to enable convenient, on demand access to a collection of user configurable computational resources (including storage, network and processors) that can be rapidly redeployed in a variety of different operating systems and hardware configurations [4]. As well as offering the end user benefits associated with ease of use, cloud computing offers considerable advantages to the institution offering such services. The main one being the reduction in costs associated with procuring and managing hardware and software resources by allowing them to be multipurpose and easy for the end user to self-manage [5].

Using cloud computing means that less physical hardware is needed to meet user demands, which positively affects the electricity usage and staff hours needed to manage these resources. This paper is organized as follows: Section 2 discusses the related works. Section 3 introduces our virtualized testbed using Openstack. A brief introduction to our prototype implementation is given in Section 4. Finally, this paper is concluded in Section 5.

ISSN: 2287-1233 ASTL

2 Related Works

OpenStack is quickly becoming the de-facto standard for IaaS Cloud platforms and is generating a lot of excitement in the research community [6]. OpenStack is a joint project between the National Aeronautics and Space Administration (NASA) and Rackspace that implements the IaaS platform via a collection of open-source cloud management packages [7]. Currently these packages include Nova (OpenStack's compute and network module), Glance (Image management module), Keystone (User management module) and Swift (Storage module) [8], [9]. One of the key advantages to OpenStack is its greater flexibility in the management of users. For example all users are assigned to Projects, which can be granted accesses to a certain percentage of the computational resources [6]. Also, unlike Eucalyptus, users can register and upload their own custom ramdisk and kernel images [7].

OpenStack is a cloud operating system that controls large pools of compute, storage, and networking resources throughout a datacenter, all managed through a dashboard that gives administrators control while empowering their users to provision resources through a web interface. OpenStack networking(aka. Neutron) that provides the virtual connectivity among the VM nodes is continuously being improved to support various network configurations and extended to support SDN/NFV requirements.

Looking at the current state-of-the-art about testbeds and tools for Cloud computing operations, it is possible to find global outfits like Amazon, which offers the EC2 [11] service, whose mission is not to support systems research but a worldwide customer base as target, with diverging needs and goals. Among other known testbeds the PlanetLab [12] initiatives (originally not conceived for Cloud computing) and the Open Cirrus Testbed [13] are the most interesting ones with regards to infrastructure resources and services provided.

The design and implementation we have discussed in this paper is easy to deploy and the cost associated with its implementation is greatly reduced due to its "free to use" software license. We submit that this paper provides useful advice to businesses that want to build their own cloud test environment and host their own services and applications. Virtualized testbed environment reduced the cost of provisioning hardware, software and lab space, yet delivered an enhanced testing environment.

3 Deployment of the Virtualized Testbed

3.1 Physical Deployment

The innovative networking technology like IDNet is allowed to be configured and experimented based on this virtualized environment [1], [2]. Our physical network nodes can be configured to VM nodes and the physical connectivity can be mapped to virtual connectivity inside the Cloud environment. The flexible virtual configuration among the network nodes allows us to try more extended network configuration and experimentation on it. It also provides easier manageability of network nodes and reproduction of test result.

We used OpenStack Kilo which is a stable version at the time of testbed deployment. Our virtualized testbed comprises VMs(Virtual Machines) for Gateway, Host, Access Router, Controller and Network/Subnetwork for each domain. For our deployment the following underlying hardware was utilized :

- The Controller Node
  : Two Intel Xeon E5-2630 2.4Ghz, 64GB Ram, Ubuntu Server 14.04 LTS
- The Compute Node
  : Two Intel Xeon E5-2630 2.4Ghz, 128GB Ram, Ubuntu Server 14.04 LTS
- The Network Node
  : Two Intel Xeon E5-2630 2.4Ghz, 64GB Ram, Ubuntu Server 14.04 LTS

The nodes communicate via a 10GB private switch. The controller node is configured to do all of the backend processes including, image management via Glance, hosting the website via Horizon and dealing with user identification via Keystone. The controller node is not running Nova compute, as the two more powerful machines are configured to be dedicated Nova compute elements.

3. 2 Virtualized Testbed Implementation for MOFI

MOFI[10] is an identity network for Future Internet supporting mobile intrinsic environment. MOFI testbed comprised of three interconnected domain networks through the KOREN research network. Each of these domain networks was designed for an HID (Host Identifier) based communication network.

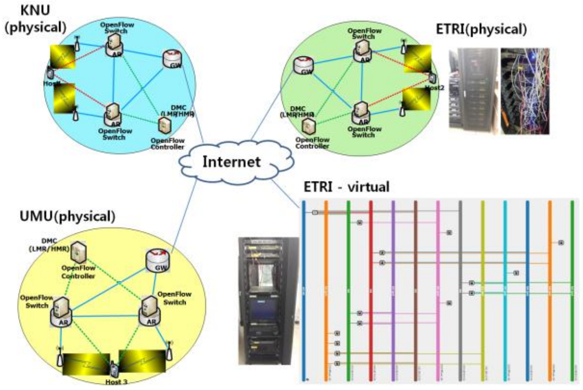

Fig. 1. MOFI Testbed

MOFI's network domain consists of three components: the Access Router (AR), MOFI Gateway (GW), and the NOX controller. These components have been deployed using OpenFlow technologies, thus implementing MOFI's control and data plane. Each one of the MOFI's gateways feature 1Gbps links for the interconnection of the three domains.

4 Experiments

The Logical network layout for the virtualized MOFI testbed environments can be seen below.

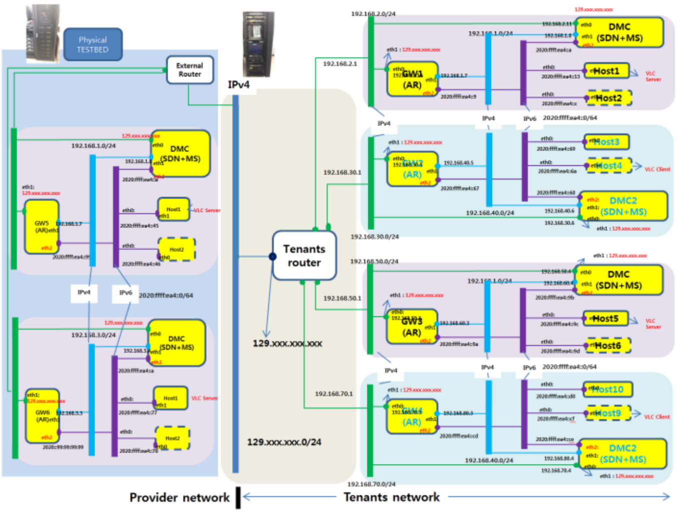

Fig. 2. Virtualized Testbed for MOFI using openstack

Virtualized MOFI testbed comprised of six interconnected domain networks. And Each network domain consisted of four components: AR, MOFI GW, NOX controller, and hosts.

Three test scenarios are performed to verify our virtualized testbed.

- Communication between VMs
  : Video streaming transmission from VLC Server(VM) to VLC client(VM).
- Communication between VM and real machine Device
  : Video streaming transmission from VLC Server(VM) to external physical device(VLC client).
- Mobility
  : Video streaming transmission from VLC Server(VM) to real device(VLC client), moving domain 1 to domain 2.

This test demonstrated the successful implementation of IaaS using an open-source cloud solution called OpenStack. Currently, we are implementing our virtualized testbed to verify the feasibility and the scalability of MOFI Functions. Our implementation is accessible at http://www.idnet.re.kr/ [6].

5 Conclusion

The cloud computing paradigm has a great potential to provide a flexible platform for delivering computational resources. Using open-source software such as OpenStack on previously purchased research hardware we have implemented an Infrastructure as a Service cloud, and investigated and explored Software as a Service solutions.

This paper was tested by constructing a virtualized testbed environment using open stack as IaaS to test the MOFI function. Virtualized testbed environment, in order to ensure interlocking and reliability with the physical test environment was configured in conjunction with the actual network. Virtualized testbed environment reduced the cost of provisioning hardware, software and lab space, yet delivered an enhanced testing environment.

References

1. IDNet, http://www.idnet.re.kr
2. Jung, H.Y., Lim, W.S. Hong, J.H., Hur, C.Y., Lee, J.C., You, T.W., Eun, J.S., Kwak, B.O., Kim, J.W., Jeon, H.S., Kim, T.W., Chun, W.J.: IDNet: Beyond All-IP Network. In: ETRI Journal Vol. 37, No. 5, October (2015)
3. Kwak, B.O., Lee, T.H., Chun, W.J.: ID Based Communication in Domain-Insulated Autonomous Network Architecture (DIANA). In: ICT Convergence (ICTC) 2012, pp.264--269 (2012)
4. Mell, P., Grance, T.: The nist definition of cloud computing. National Institute of Standards and Technology (NIST), Special Publication (2011)
5. Vaquero, Luis M. Rodero-Merino, Luis. Caceres, Juan and Lindner, Maik. : A break in the clouds: towards a cloud definition. In: SIGCOMM Computer Communication, Vol.39(1), pp.50--55 (2008)
6. Laszewski et al.: Comparison of multiple cloud frameworks Cloud Computing (CLOUD), In: IEEE 5th International Conference on, pp.734--741 (2012)
7. Vockler et al.: Experiences using cloud computing for a scientific workflow application ACM (2011)
8. LLC OpenStack. Openstack install and deploy manual (2012)
9. Zhang, Jilin., Zhang, Jianhui., Ding, H., Wan, J., Ren, Y., Wang, J.: Designing and applying an education iaas system based on openstack. Appl. Math, 7(1L), pp.155--160 (2013)
10. MOFI, http://www.idnet.re.kr/mofi/
11. Amazon EC2, http://aws.amazon.com/ec2
12. Peterson, T. R. TL., Muir, S., A. Klingaman, A.: Planetlab architecture: An overview. PlanetLab Consortium, Tech. Rep. PDN-06-031 (2006)
13. Avetisyan, A., Campbell, R., Gupta, I., Heath, M., Ko, S., Ganger, G., Kozuch, M., O'Hallaron, D., Kunze, M., Kwan, T., Lai, K., Lyons, M., Milojicic D., Lee, H., Soh, Y., Ming, N., Luke, J.-Y, Namgoong, H.: Open cirrus: A global cloud computing testbed. In: Computer, vol. 43, no. 4, pp.35--43 (2010)