The Study on Configuration of Multi-Tenant Networks in SDN Controller

Y.Y.SHIN*, S.H.KANG*, J.Y.KWAK*, B.Y.LEE*, S.H.YANG*

* ETRI(Electronics and Telecommunications Research Institute), Korea

uni2u@etri.re.kr, skang@etri.re.kr, jiyoung@etri.re.kr, bylee@etri.re.kr, shyang@etri.re.kr

Abstract— **The Virtual Tenant Network (VTN) is provides multi-tenant virtual network on an SDN controller. Conventionally, huge investment in the network systems and operating expenses are needed because the network is configured as a silo for each department and system.**

Keywords—**VTN(Virtual Tenant Network), SDN, OpenFlow, Virtual Network, Controller**

I. Introduction

Multi-tenant networks, in a nutshell, are data-center networks that are broken up and logically divided into smaller, isolated networks. Like tenants in an apartment complex, multi-tenant networks... share the physical networking gear but operate on their own network without any visibility into the other logical networks. While the capability to separate networks into logical units has been available for some time through the use of VLANs, virtualized data-centers and cloud computing concepts have brought multi-tenancy back to the attention of network administrators.

Whether defined by the business processes of the organization or federal regulatory requirements, the need to isolate and control parts of the network does not change when moving to the cloud. Taking the multi-tenant network approach that public cloud providers use can ensure that local data is secure and may even help evolve the IT organization into one focused on services and offerings, rather than simply bits and bytes.

II. Multi-tenant Network

As enterprises transition from traditional dedicated server deployments to virtualized environments that leverage public cloud, private cloud and hybrid cloud services, the cloud computing networks they are building must provide security and segregation of sensitive data and applications. Network architects may find a solution in building a multi-tenant network.

The need for isolation within public cloud offerings is clear: Customers pay for the amount of bandwidth and capacity they receive and trust that the vendor will not expose their information to any other customer.

A. Needs

The first cloud computing applications were delivered as software as a service (SaaS) and segregated user data programmatically by user accounts or separate databases.

However, as cloud-based solutions have moved up the stack to include both platform as a service (PaaS) and now infrastructure as a service (IaaS) , segregation is required not only in data sets but on network components themselves.

B. Private Cloud

The role of multi-tenant networks in the private cloud is not as obvious but is equally important. Enterprises often have specific regulatory or business requirements, such as HIPAA or PCI compliance, which demand that a given application or service be isolated. Along with the security aspects of multi-tenancy networks, breaking the data-center network by application and service could be the springboard to evolve the IT organization from an operations mode into essentially an internal vendor, providing applications and services to business units in much the same vein as public cloud service providers. As with service providers, building a private cloud infrastructure would enable enterprise applications to be delivered to business units as a service, with clearly defined service-level agreements and bill-back procedures. Multi-tenant networks support this effort by allowing specific quality of service and security policies to be set for each "customer" of IT services.

III. Configuration of Tenant

There are many ways to define to Multi-Tenant Networking. However, there is no definition for security.

The several different options for specifying what machines are part of a specific Virtual Network Segments. These include:

- Location information such as the switch and interface to which a machine is attached.
- Packet information such as the layer 2 MAC address, the layer 3 IP address, or VLAN.
- External information that can be mapped to either location or packet information.
- Any combination of the above.

These policies are defined by using "segment-rule" commands in the RESTAPI, and each member of the Virtual Network Segment is considered attached to an interface in the Virtual Network Segments.

In this paper, we define a multi-tenant to discuss three ways.

A. Tenants

The SDN controller application allows you to take a pool of compute resources (e.g., virtual machines) and allocate them to different groups, called tenants.

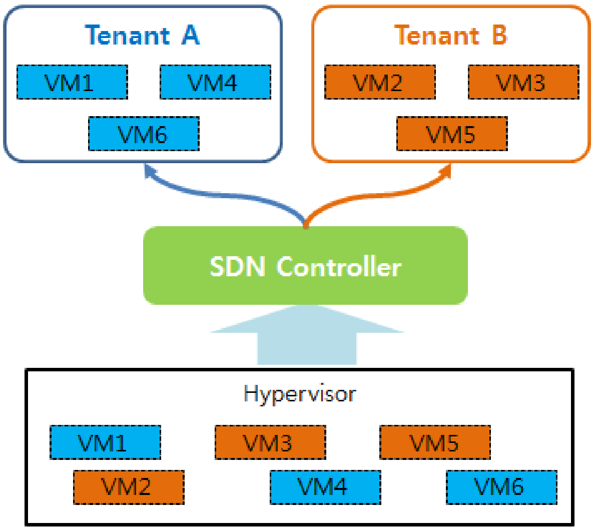

Figure 1. Configuration Tenants

B. Segments

Further, the VMs within a tenant can be further grouped into logical, virtual networks. Each of these is called a "virtual network segment".

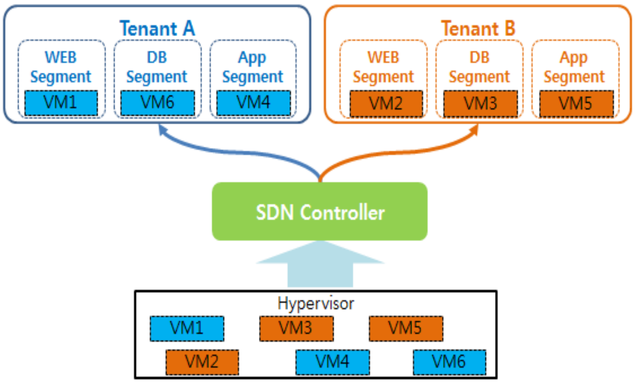

Figure 2. Configuration Tenants and Segments

1) ***MAC Address and Cost Policy Based:*** MAC Address is a unique name for the device. Using each device's MAC defines tenant network. So many Tenant solutions are based MAC address. Network managers are aware of all devices MAC address.

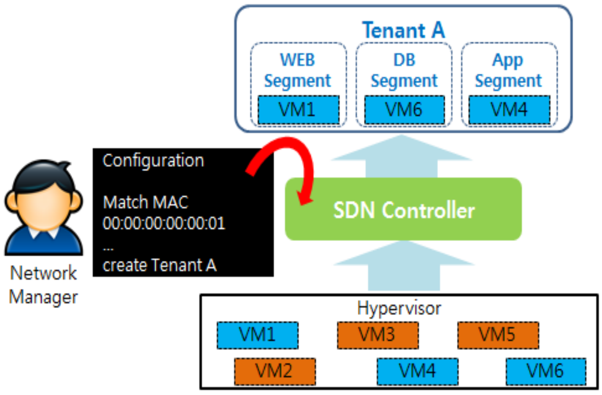

Figure 3. Configuration Segment using MAC Based.

Tenant to configure the network MAC address and cost-use policies for cost comprise the virtual network.

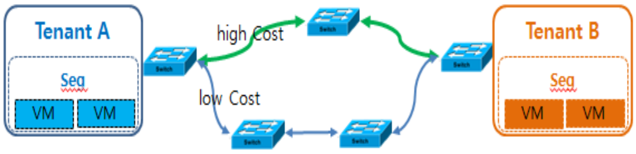

Figure 4. Configuration MAC and Cost Policy Based.

2) ***IP-subnet and Cost Policy Based:*** Tenant based on the IP address can be configured. The Controller is known all devices information. So matching Devices information from Controller. But configured unknown devices, Controller can't find matching devices. And same IP Address is too.

Configuration unknown Device case is a no problem. Because Tenant network is not affected when create same VMs.

But configuration same ip case is create big problem. Controller based on the ip cannot match the actual device. So Controller needs a find device Function. This function can notification to Network Manager.

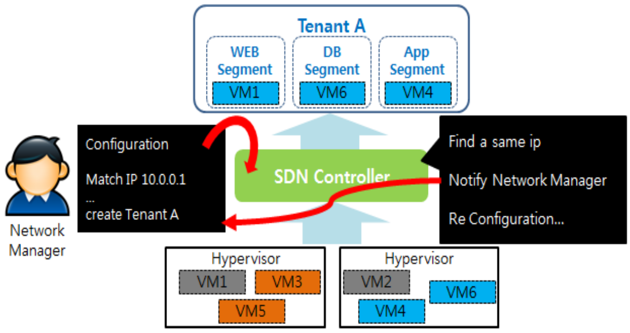

Figure 5. Configuration same private IP in Tenant

The administrator can assign the same Private IP in Cloud to Cloud networks.

We can find a right device using parents Information. The Tenant is a parent of Segment.

(e. g., Tenant A is a parent of WEB Segment(VM1), Tenant B is parent of WEB Segment(VM2). Same IP address VM1, VM2, Figure 6)

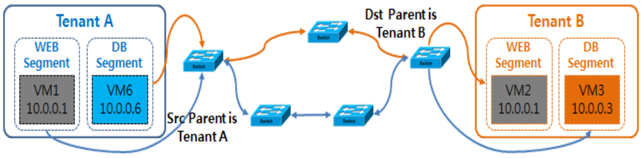

Figure 6. Flow of same private IP in Tenant

3) ***VLAN and Cost Policy Based:*** Tenant based on the VLAN ID can be configured. This is the traditional method. We can configuration VLAN in OpenFlow Switch.

IV. TENANT AND SEGMENT ISOLATION

A. *Virtual Router*

Virtual routing is responsible for finally determining the actual policy that should apply to the flow.

The policy for packets is determined examining the VNS assignments for the source and the destination of the packet, and applying policy based on the configuration for the virtual network segments.

The application allows you to control connectivity and traffic patterns in several different areas.

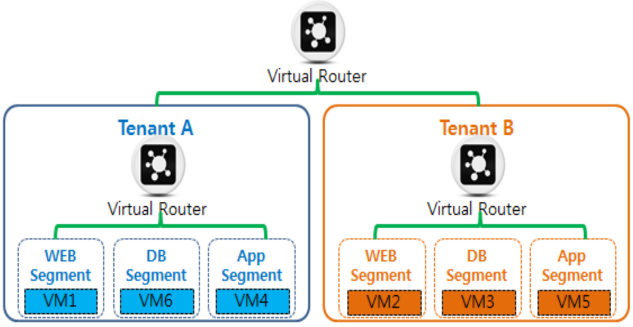

Figure 7. Each Tenant create Virtual Router

- Between hosts within a given virtual network segment.
- Between different virtual network segments of a given tenant.
- Between virtual network segments of different tenants.
- Between hosts, virtual network segments, tenants, and the external network.

Distributed virtual routing allows you to define connectivity among different Virtual Network Segments as well connectivity between Virtual Network Segments hosts and the external network.

Distributed virtual routing is achieved through a set of distributed virtual routers. Each tenant will have its own distributed virtual router to define the connectivity among the Virtual Network Segments under the same tenant.

A distributed virtual router is conceptually a single entity, but it is implemented across all the OpenFlow switches in the network. There is no single routing instance running on a single machine/hypervisor that is all Virtual Network Segments traffic must route through. The policy described in the distributed virtual router is pushed down to all the required switches under the SDN Controller. This allows for extremely efficient and scalable performance and policy enforcement.

In addition to this there is a system-wide distributed virtual router which connects different tenant routers and defines the connectivity among different tenants and to the outside world.

1) ***Routing Between Segments:*** Once router interfaces are created and connected, you can specify the routing rules on the distributed virtual router.

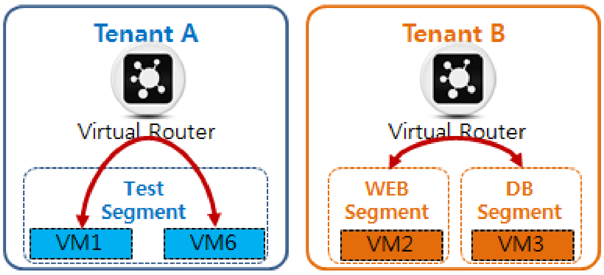

Figure 8. Routing Between Segments

2) ***Routing Between Tenants:*** The 'system' tenant and its distributed virtual router, 'vrsystem', control connectivity among different tenants. Configuring the system distributed virtual router is similar to configuring a tenant's distributed virtual router. The main difference is that the routes are between tenants instead of between Virtual Network Segments.

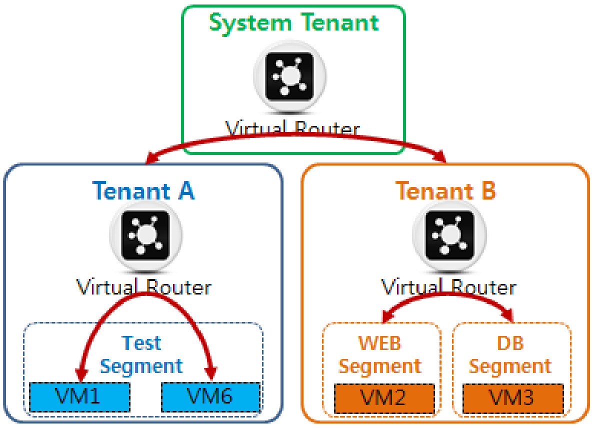

Figure 9. Routing Between Tenants

3) ***Routing to External Network:*** To communicate among different tenants and communicate with outside world, we need to configure system tenant/system virtual router and external tenant accordingly.

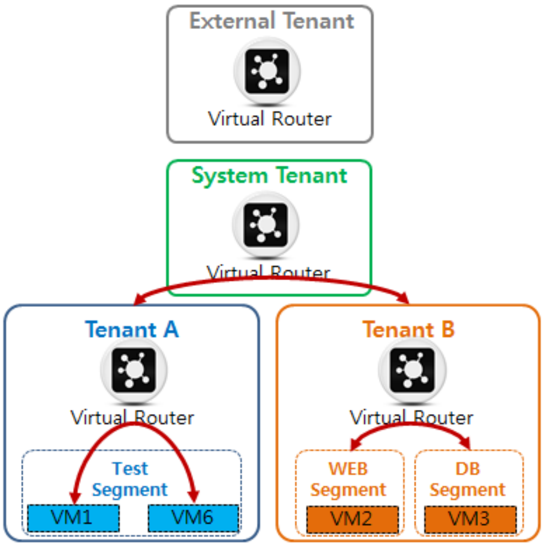

Figure 10. Routing to External Network

V. Conclusions

Depending on the data center environment, any of these methods or a mix of them will achieve the stated goal of creating a multi-tenant network to support private cloud deployments. The network is ultimately still just pushing packets, however, with very little context for the applications and data that is passing through. The rise of cloud computing concepts and the issues with multi-tenancy only serve to dispel the notion held by many that data-center network switches only have to enable "big, dumb pipes," providing high performance and bandwidth without concern for the traffic itself. As data-centers become more focused on applications and service, the network switches and gear that reside within those data-centers must become focused on more than pushing packets around.

For the network to be application aware, it must rely on knowing where a particular packet is coming from and going to, and it must understand the characteristics and requirements of the application that is pushing that packet. The application would ultimately dictate the latency and bandwidth requirements, for example, and an application-aware network could then dynamically adjust itself to meet those demands. In an application-aware network, the private cloud and all of the applications would become the tenants on the network, receiving the appropriate isolation and security based on each application's specific requirements.

Acknowledgment

This research was funded by the MSIP (Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2013.

References

[1] Robbie Higgins, vice president of security services at GlassHouse Technologies, Securing a multi-tenant environment
[2] Primer: Multi-tenant network for the private cloud, August, 2010
[3] NEC, NEC contribution to OpenDaylight: Virtual Tenant Network (VTN), June, 2013
[4] NEC, ProgrammableFlow Networking:The Simple Solution for Complex Networks
[5] BigSwitch Networks, Big Virtual Switch and OpenStack
[6] Pino de Candia, midokura, Overlay-based virtual networking vs OpenFlow-controlled switch fabrics in IaaS Clouds, November, 2012
[7] Renato Recio, IBM, Distributed Overlay Virtual Ethernet(DOVE) Networks, 2012