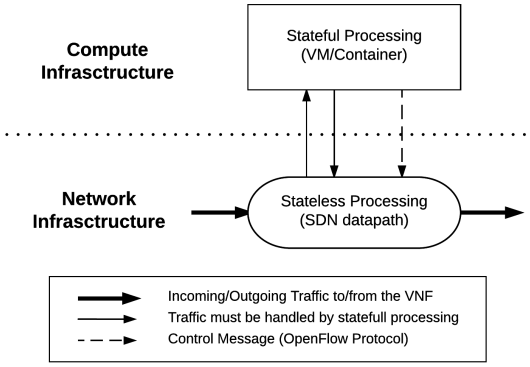

Fig. 1. vCPE Service Design Architecture

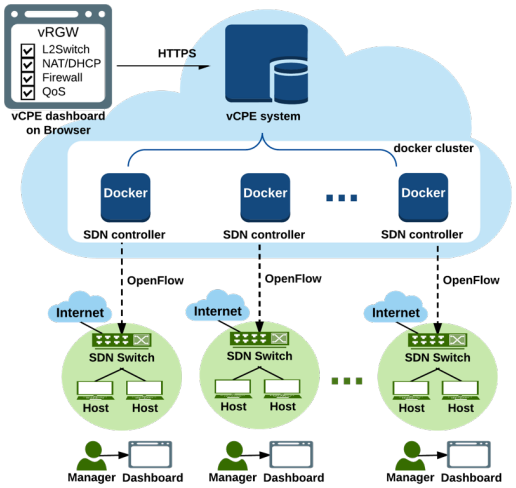

Fig. 2. Service Deployment Model

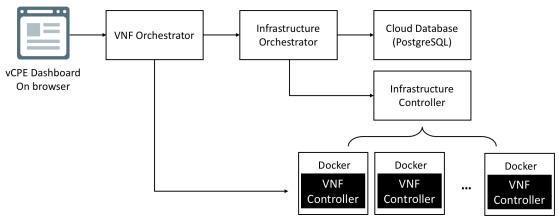

Fig. 3. Service Syetem Overview

I. vCPE System Overview

Our object is to demonstrate that our vCPE system can help the telco to reduce the Capex and Opex when deploying, maintaining and updating CPE service. This section describes our vCPE system architecture and how we achieve our goal. In the subsection I-A, we will introduce the overview architecture of our virtual CPE Service. Then in subsection I-B, we will explain that through the advantages of service architecture, our vCPE system can deploy our service to the customer easily. In the subsection I-C, we will give details of the whole vCPE system.

A. vCPE Service Design Overview

With the concept of SDN-enabled [1] VNFs in Fig. 1, the network functions have been achived by the synergies between compute and network infrastructures. The former is a VNF controller, mainly responsible for dealing with stateful processing. The latter is a SDN switch, used for stateless processing.

1) Stateful Processing component (VNF controller in container): This component have to perform more complex algorithm, keep the state associated with the VNF and provide interface for service providers or customers to configure and update the behavior of the stateless datapath processing component, since software is good at these tasks. We use SDN controller to implement the NFV controller and it's worth noting that we use southbound APIs of SDN controller to handle the interface between the stateful and stateless component with OpenFlow protocol, which was originally designed for this.

2) Stateless Processing component (SDN datapath): Stateless processing component, are implemented by SDN datapath resources, which is optimized for data plane traffic processing. Since SDN datapath have decoupled the control plane and data plane, so it can accept the control message from the stateful processing component.

Using the advantages of this architecture, we can assign stateless or light-weight state work to the SDN switch, for example, packet filtering and packet counting, to load-off the computing resources. If we want to update our service, we just need to update the statful component, since the stateless component just follow the command from stateful components.

B. Service Deployment Model

With architecure mentioned in I-A, we come up with a network function service deployment model. Because the algorithm and policies handled by computing infrastructures, and the generic network devices, that is, SDN switches, only do stateless processing. The customer only need to buy a general SDN switch at their home gateway and subscribe different NFV controller through our vCPE platform, they will have a different network function service.

In figure 2 illustrated the service deployment model, each green area is a local network domain of customer. At the gateway of this domain, there's a SDN-enabled switch. The customer can subscribe to our vcpe service through our vcpe dashboard. After subscribing, the vCPE system will create a new docker container, in which running a SDN controller we developed. The customer only need to setup the gateway SDN switch to connect the SDN controller by the OpenFlow protocol, then the switch will handle these service.

C. Architecture of the vCPE system

1) System Overview: The architecture as shown in Fig 3 including of a Infrastructure Controller, Infrastructure Orchestrator, Cloud Database, VNF Controllers and VNF Orchestrator.

The Infrastructure Orchestrator, VNF Orchestrator and Cloud Database are web servers running on virtual machines hosting in the cloud using Amazon Web Services (AWS) Data Center also the Infrastructure Controller is a powerful physical blade server running in our laboratory. Each component is introduced in the subsection below.

2) Infrastructure Controller: The infrastructure controller is a composed Docker management server with the ability to manage the Docker resources like containers and images. The infrastructure does'nt handle the customer authentication or maintaining the state of running service, it just follow the request from the infrastrcure orchestrator to create, delete, start, stop and inspect containers.

3) Infrastructure Orchestrator: The infrasctructure orchestrator plays the key role of our system. It connecting and automating of workflows when we deploy our services. When a customer subscribes, the infrasctructure orchestrator authenticates the customer first, next it will call the infrastructure controllerto create a container for this customer, and update information in database afterwards. It handle the entire lifecycle of our vCPE service.

4) Cloud Database: The cloud database is used for restoreing the of our vCPE services, which include each customer's credential, customer's container settings and virtual CPE service states. The cloud database is using PostgreSQL, which is a open source, easily customized and object-relational database system. Only Infrastructure Orchestrator has read and write permissions to access cloud database.

5) VNF Controllers: VNF Controllers contains a SDN controller developed with ryu framework and a remote launcher module. The SDN controller does not have a remote launcher module to remotely execute a SDN controller. We built a lightweight server as a launcher module to resolve the remotely execution issue. The remote launcher module monitor the SDN controller process ID (PID) and properly kill the SDN controller process ID when on demand. When the infrastructure controller once create the container, the remote module will run up initiallly, waiting request from VNF VNF Orchestrator. The details of SDN controller desgin will be presented at section II

6) VNF Orchestrator: The VNF Orchestrator is a Web application server hosting on Amazon Web server, being online for customer and provide a dashboard for virtual CPE and containers management and configuration.

Through the web UI provided by the VNF Orchestrator, the customers can subscribe to the desired service and without typing any command via the command line interface (CLI). After receiving the subscribing message, the VNF orchestrator will request the infrastructure orchestrator to create a new VNF controller, and then send the virtual CPE configuration to the new VNF controller. Based on configuration demands under different conditions, the network administrator is able to select any of the listed services on the dashboard such as NAT, DHCP, Firewall and QoS management.

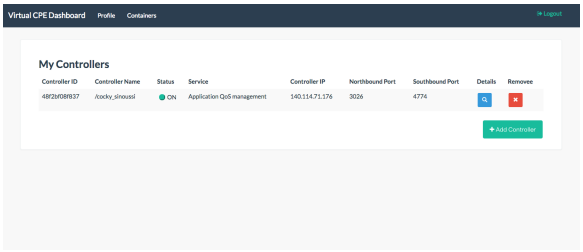

Fig. 4. Virtual CPE Dashboard User Interface for Container Management

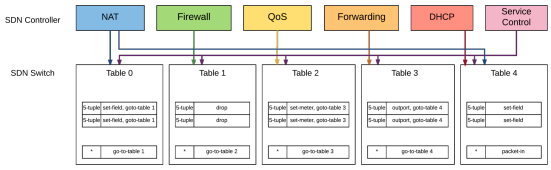

Fig. 5. The flow table order of our vCPE service

II. vCPE Services

A. Multiple Flow Flow Tables Strategy

In subection I-A, the vCPE service design architecture have been introduced. The network functions are handled by the cooperation between SDN controller on cloud and SDN switch at the local network gateway. The controller transform the network functions to series of OpenFlow rules and send rules requests to SDN switch. Following the command from controller, the SDN switch insert rules to flow tables, checks incoming packets against the flow entry match fields, excute the actions in matching rules. The flow tabe defines all matching and processing, playing an important role to executive network function. Since the restricttion of single table have been discussed in [2], we use multiple flow tables strategy to implement our vCPE service.

In multiple flow tables strategy, the most important question is: which flow table should we insert rules into? We use the network function as a demarcation, that is, SDN applications which are responsible for specific network functions will only insert rules to one specific flow table. So we can focus on the design of the network function itself. In this way, howerver, the order of flow table become crucial. Should we put flow table of this network function at first, or the other? The answer is about the type of match and action in the rules generated by the network function.

The network functions of our vCPE services including Forwarding, Firewall, NAT, DHCP and QoS. We have determined the order of each function, shown in 5. In the following subsections, we will introduce how to implement these network functions, which type of rules will be inserted to SDN switch, and how these rules affect our decistion of the order of flow tables.

B. Forwarding

C. Firewall

D. NAT

E. DHCP

F. QoS

REFERENCES

[1] J. Matias, J. Garay, N. Toledo, J. Unzilla, and E. Jacob, "Toward an SDN-enabled NFV architecture," *IEEE Communications Magazine*, vol. 53, pp. 187–193, apr 2015.

[2] O. N. Foundation, "The benefits of multiple flow tables and ttps," tech. rep., 2015.