

Sistema operacional ONOS aplicado à SDN/NFV

Fernando Henrique Santorsula¹

¹Universidade Estadual de Campinas - (UNICAMP)
Av. Albert Einstein, Nº 400 - Cidade Universitária, Campinas - SP, 13083-852

²Faculdade de Engenharia Elétrica e de Computação - (FEEC)
f208918@dac.unicamp.br

Abstract. *This article will describe the entire history, installation, connectivity tests and features of the ONOS network operating system (Open Network Operating System), which is an extremely robust and versatile SDN controller, used by several companies and educational institutions around the world, working for new projects and research, still in this article, concepts and characteristics about SDN and NFV will be addressed.*

Resumo. *Este artigo irá descrever toda a história, instalação, testes de conectividade e recursos do sistema operacional de rede ONOS (Open Network Operating System), que é um controlador SDN extremamente robusto e versátil, utilizado por diversas empresas e instituições de ensino ao redor do mundo, atuando para novos projetos e pesquisas, ainda neste artigo, será abordado conceitos e características sobre SDN e NFV.*

1. Introdução ao ONOS

O projeto do sistema ONOS foi iniciado em 2012 pela liderança de Pankaj Berde (arquiteto de software do Open Networking Lab), o sistema ONOS foi oficialmente lançado em 2014 pelo Open Networking Lab, em parceria com algumas grandes empresas, tais como: AT&T, Cisco, NEC, Verison, NTT Communications, entre outras. O ONOS nasceu totalmente com seu código aberto e foi liberado para a The Linux Foundation, que se juntou ao projeto, desenvolvendo melhorias e correções.

O ONOS - (Open Network Operating System), é um sistema distribuído ou seja! uma coleção de dispositivos autônomos conectados por uma rede de comunicação, o ONOS foi desenvolvido para atuar como controlador SDN (Software Defined Networks), o ONOS foi projetado especificamente para atender a escalabilidade e alta disponibilidade de ambientes de redes SDN.

O ONOS se comporta como um sistema operacional de rede, fazendo a separação do plano de controle e dados, podendo ser utilizado em redes de longa distância e para provedores de serviços que são transportados em grandes redes. O ONOS é um software totalmente open source, mantido pela The Linux Foundation.

2. Características sobre a instalação do ONOS

O ONOS pode ser facilmente instalado e utilizado na arquitetura x86, entre outras, em especial em switch's white box, atuando na comutação de camada 2 e camada 3, em ambientes pequenos e grandes como um data center, o ONOS se destaca com o uso de hardware "aberto", os mais utilizados são os da empresa Lanner que pode contar com vários modelos de switch, roteadores, servidores, podendo ter interfaces de

1 Gbe, 10 GbE e 40 GbE, aplicado para alta disponibilidade, alto desempenho e excelente custo-benefício, incluindo soluções de sobreposição de virtualização de rede que funcionam extremamente bem com controladores SDN como ONOS, para ter êxito na instalação, basta seguir os requisitos que estão presentes na tabela 1º

Sistema operacional utilizado: Ubuntu Server LTS – 16.04
Configuração de Hardware:
Processador Intel Core i5
2GB de memória RAM
1 NIC (Modo Brigde)
50GB de HD
Plataforma de instalação: Virtual (Oracle VirtualBOX – Versão: 5.2.42)

Tabela 1º

3. Procedimento de instalação

a-) Atualizar o sistema, seus pacotes e reiniciar o sistema:

```
apt update && apt upgrade -y && reboot
```

b-) Instalar os pacotes auxiliares: Nmap (Snnifer), Editor de textos (VIM), Ferramentas de Redes (Net-tools), Monitor de Sistema (Htop):

```
apt install nmap vim net-tools htop -y
```

c-) Instale os pacotes que serão utilizados pelo ONOS

```
apt-get update && apt-get install maven git openjdk-8-jre openjdk-8-jdk unzip -y
```

d-) Faça o download da versão 1.12.0 do ONOS e depois descompacte o arquivo:

onos-1.12.0.tar.gz

```
wget https://itfusion.com.br/onos/onos-1.12.0.tar.gz
```

```
tar xzf onos-1.12.0.tar.gz
```

e-) Acesse o diretório bin do ONOS e execute os seguintes comandos:

```
cd onos-2.0.0/bin/
```

```
export JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-amd64
```

```
./onos-service
```

f-) Após abrir o terminal do ONOS, execute o comando abaixo para ativar as aplicações:

```
apps -a -s
app activate org.onosproject.fwd
app activate org.onosproject.openflow
```

g-) Acesse o ONOS pela interface web com seu determinado endereço de IP, exemplo:

<http://SEU-IP:8181/onos/ui/login.html>

<http://192.168.2.111:8181/onos/ui/login.html>

Observações:

Neste artigo, o endereço de rede do servidor ONOS e Mininet são:

ONOS: 192.168.2.111/24

Mininet: 192.168.2.110/24

Este usuário e senha são definidos de forma default, após a instalação do ONOS.

Usuário: onos

Senha: rocks



User:

Password:

Login

Observação 1º:

Para testes de conectividade, no próximo tópico **h-)** será necessário ter uma OUTRA máquina virtual ou física com o Mininet instalado, caso não tenha o Mininet instalado, basta acessar o link abaixo, onde irá conter um tutorial de instalação do Mininet, para concluir os testes práticos com o controlador SDN “ONOS.”

Link de instalação do Mininet:

http://fhs.pro.br/?page_id=2816

h-) Acesse a máquina (Mininet) – 192.168.2.110 e execute os comandos a seguir:

```
sudo mn --controller=remote,ip=192.168.2.111
```

```
h1 ping h2 -c 10
```

```
h2 ping h1 -c 10
```

Observação 2º:

Cada comando deverá ser executado de forma única e pressionando ENTER para confirmar cada operação, conforme as próximas ilustrações.

```
root@mininet:~# sudo mn --controller=remote,ip=192.168.2.111
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet> █
```

```

mininet> h1 ping h2 -c 10
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=9.11 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.548 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.097 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.097 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.084 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.098 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.089 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.103 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.075 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.091 ms

--- 10.0.0.2 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9000ms
rtt min/avg/max/mdev = 0.075/1.039/9.115/2.695 ms
mininet> h2 ping h1 -c 10
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.051 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.116 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.091 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.090 ms
64 bytes from 10.0.0.1: icmp_seq=5 ttl=64 time=0.100 ms
64 bytes from 10.0.0.1: icmp_seq=6 ttl=64 time=0.091 ms
64 bytes from 10.0.0.1: icmp_seq=7 ttl=64 time=0.096 ms
64 bytes from 10.0.0.1: icmp_seq=8 ttl=64 time=0.100 ms
64 bytes from 10.0.0.1: icmp_seq=9 ttl=64 time=0.121 ms
64 bytes from 10.0.0.1: icmp_seq=10 ttl=64 time=0.092 ms

--- 10.0.0.1 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9101ms
rtt min/avg/max/mdev = 0.051/0.094/0.121/0.021 ms
mininet>

```

The screenshot displays the ONOS (Open Network Operating System) web interface. At the top, the ONOS logo and name are visible, along with the tagline "Open Network Operating System" and a search icon. The main area shows a network topology with a central switch (blue square) connected to two hosts (yellow circles) labeled 10.0.0.1 and 10.0.0.2. A sidebar on the left shows a selected host with IP 192.168.2.111 and 1 device. On the right, an "ONOS Summary" panel provides key statistics:

ONOS Summary	
Version :	1.12.0
Devices :	1
Links :	0
Hosts :	2
Topology SCCs :	1
Intents :	0
Tunnels :	0
Flows :	7

Below the summary, a specific flow is identified with the URI: of:000000000000000001. Further details for this flow are listed:

URI :	of:000000000000000001
Vendor :	Nicira, Inc.
H/W Version :	Open vSwitch
S/W Version :	2.5.5
Serial # :	None
Protocol :	OF_13

At the bottom of the interface, there is a toolbar with various icons for network management, including a search icon, a plus sign, and a refresh icon.

4. Aplicações do ONOS

O ONOS possui inúmeras aplicações que podem ser utilizadas para simular novas topologias de redes que utilizam a arquitetura SDN, as mesmas iremos abordar mais adiante neste artigo, para ficar mais simples a compreensão, ilustramos um ambiente corporativo, onde possui uma solução de Segurança da Informação, mais precisamente um “Firewall”, onde o mesmo será ilustrado no modo “tradicional” e no modo “controlador”, utilizando o ONOS, vejamos:

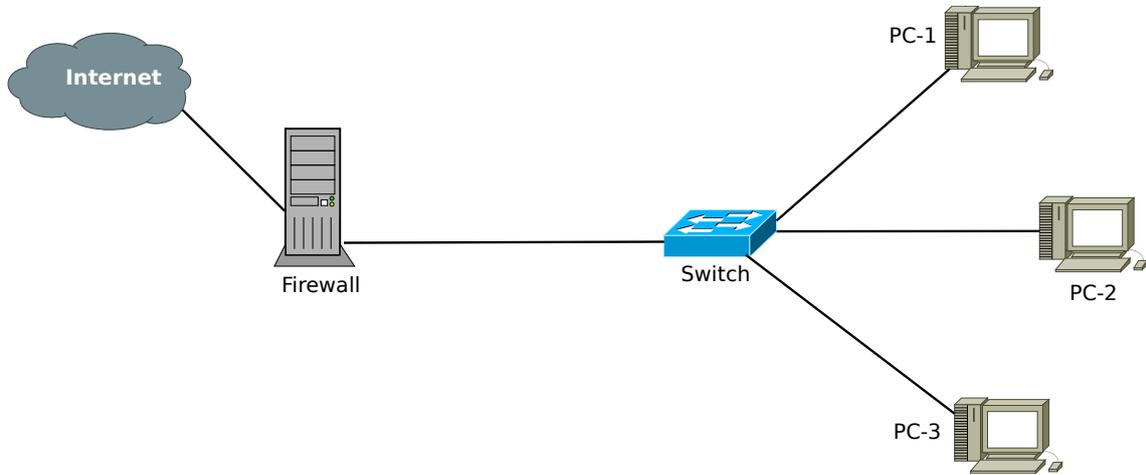


Figura 1

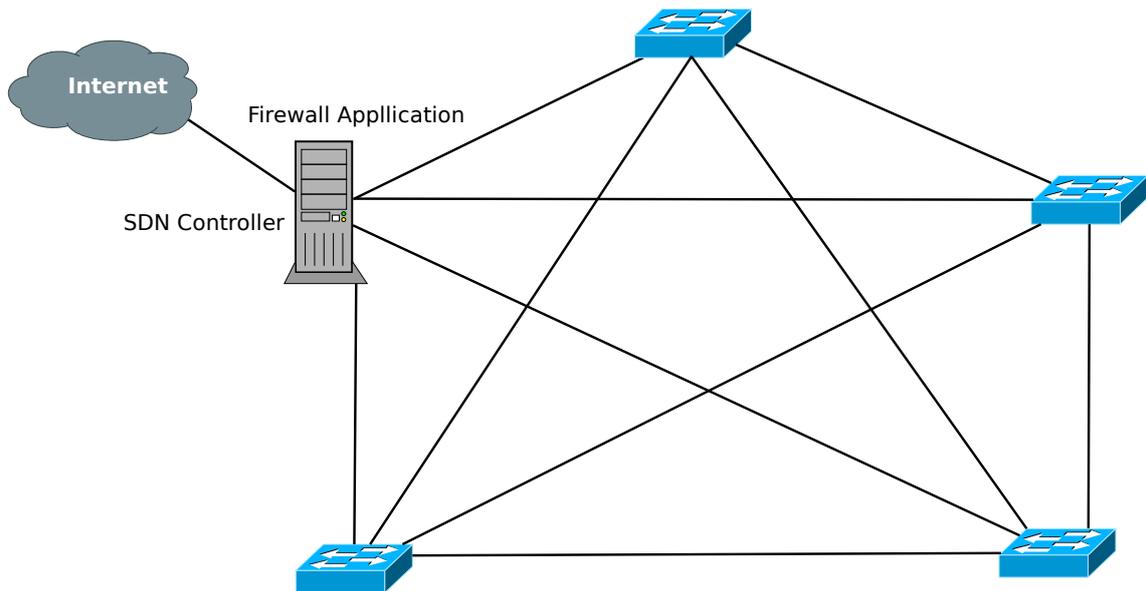
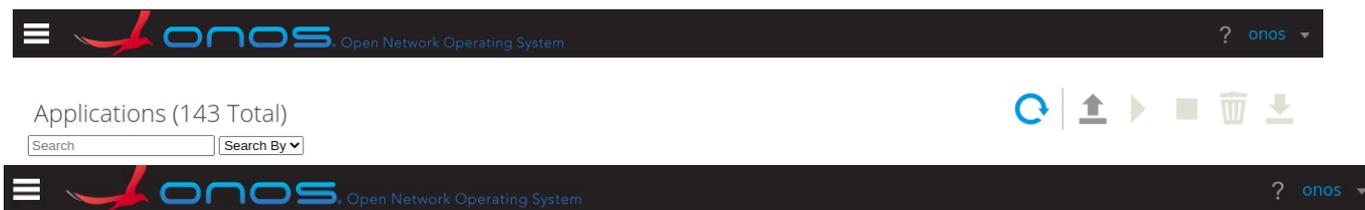


Figura 2

5. Aplicações

Ao acessar a interface web do ONOS, podemos ver as inúmeras aplicações que podemos utilizar com este magnífico controlador, na versão versão do ONOS: 1.12.0, a mesma possui 143 aplicações, conforme podemos ver na figura 3, onde as que estão habilitadas em verde, estão ativos no ONOS.

Figura 3



Cluster Nodes (1 Total)

ACTIVE	STARTED	NODE ID	IP ADDRESS	TCP PORT	LAST UPDATED	
✓	✓	192.168.2.111	192.168.2.111	9876	7:12:14 PM -03:00	
✓	📄	Optical Network Model	org.onosproject.optical-model	1.12.0	Optical	ONOS Community
✓	📄	Reactive Forwarding	org.onosproject.fwd	1.12.0	Traffic Steering	ONOS Community
	📄	Access Control Lists	org.onosproject.acl	1.12.0	Security	ONOS Community
	📄	Arista Drivers	org.onosproject.drivers.arista	1.12.0	Drivers	ONOS Community
	📄	Artemis	org.onosproject.artemis	1.12.0	Monitoring	ONOS Community

O ONOS por padrão possui inúmeros recursos de fácil acesso, auxiliando o gestor da aplicação fazer configurações rápidas e visualizar recursos em atividade, conforme a ilustração a seguir, podemos visualizar o nó principal e ativo no ONOS.

Figura 4

Nesta área do controlador, podemos visualizar de forma detalhada a quantidade de dispositivos que temos em no controlador ONOS, em nosso ambiente temos um equipamento (Switch) do fabricante Nicira, Inc. em operação:

FRIENDLY NAME	DEVICE ID	MASTER	PORTS	VENDOR	H/W VERSION	S/W VERSION	PROTOCOL
✓ 📄 of:000000000000000001	of:000000000000000001	192.168.2.111	3	Nicira, Inc.	Open vSwitch	2.5.5	OF_13

Figura 5

Nesta outra área do controlador, podemos visualizar de forma detalhada a quantidade de hosts que temos conectados em nosso controlador, neste caso temos 2 equipamentos conectados e que testamos sua conectividade com o Mininet:

FRIENDLY NAME	HOST ID	MAC ADDRESS	VLAN ID	CONFIGURED	IP ADDRESSES	LOCATION
📄 10.0.0.2	42:6F:1F:24:41:46/None	42:6F:1F:24:41:46	None	false	10.0.0.2	of:000000000000000001/2
📄 10.0.0.1	66:A4:38:AD:2E:F7/None	66:A4:38:AD:2E:F7	None	false	10.0.0.1	of:000000000000000001/1

Figura 6

6. Hardware Open Source

O ONOS possui suporte nativo à vários fabricantes de hardware aberto, separamos um modelo específico, onde o ONOS pode ser facilmente instalado, este equipamento é desenvolvido pela empresa Lanner, que é uma das referências mundiais em hardware aberto, veja um exemplo ilustrado de equipamentos da empresa Lanner, mas figuras 7 e 8.



Figura 7

NCA-2512: Expanding Lanner's High-availability SDN/NFV Hardware Solutions



Figura 8

7. Introdução a SDN

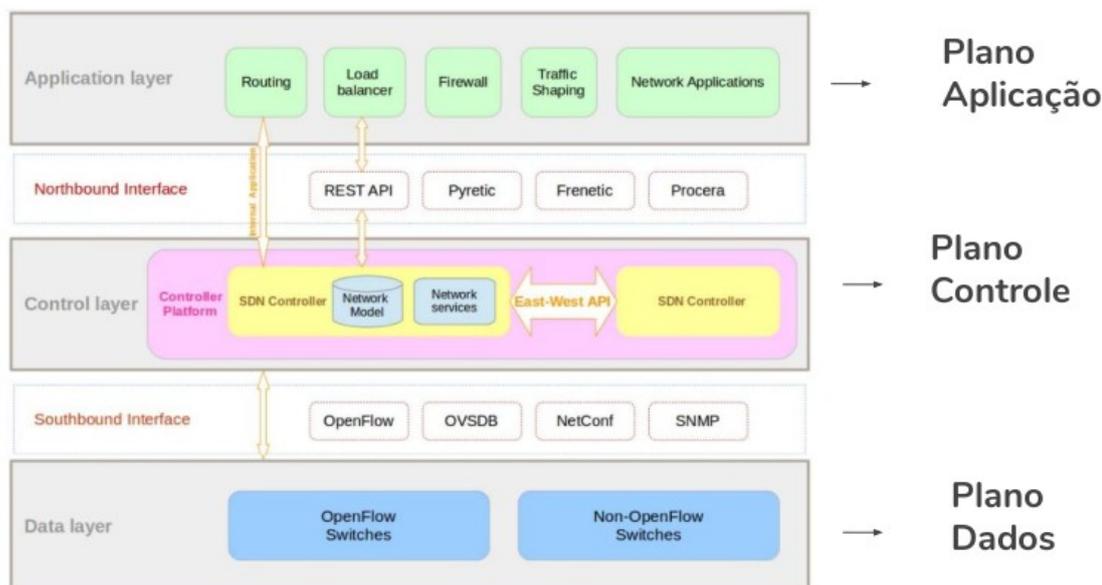
Nas atuais redes de computadores empresariais, instituições sem fins lucrativos e afins, possui a finalidade de compartilhar recursos, transportar informações de uma origem até um destino, sendo assim, as redes computacionais, possuem camadas embutidas nos dispositivos de rede, com a finalidade de melhorar a flexibilidade, inovação e evolução tecnológica, entre outros fatores positivos para o negócio ou pesquisa.

Porém esses dispositivos possuem configurações que devem ser feitas manualmente por profissionais especializados, utilizando um software do fabricante, ou seja! Uma camada de controle decide o que fazer com os pacotes de informação, gestão de tráfego e as demais decisões à ser tomada em determinada rede ou conjunto de redes computacionais.

O SDN (Software Defined Networking), possui um paradigma diferente do tradicional, onde existe uma separação da camada de controle, da camada de dados, melhorando os recursos de uma ou mais redes, simplificando o gerenciamento, reduzindo os custos de operação, melhorando a inovação e evolução tecnológica.

A arquitetura do SDN fica em um nível mais baixo, que é representada pela infraestrutura ou camada de dados, onde é nesta camada que os dispositivos encaminham o tráfego baseado nas decisões da camada de controle. A comunicação de

dados e a camada de controle pode ser utilizada por vários protocolos, tais como: OpenFlow, SNMP, BGP, NetConf entre outros, na figura 9 podemos compreender melhor o novo conceito de redes definidas por software.



Imagens do artigo: Distributed SDN Control: Survey, Taxonomy, and Challenges
 IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 20, NO. 1, FIRST QUARTER 2018

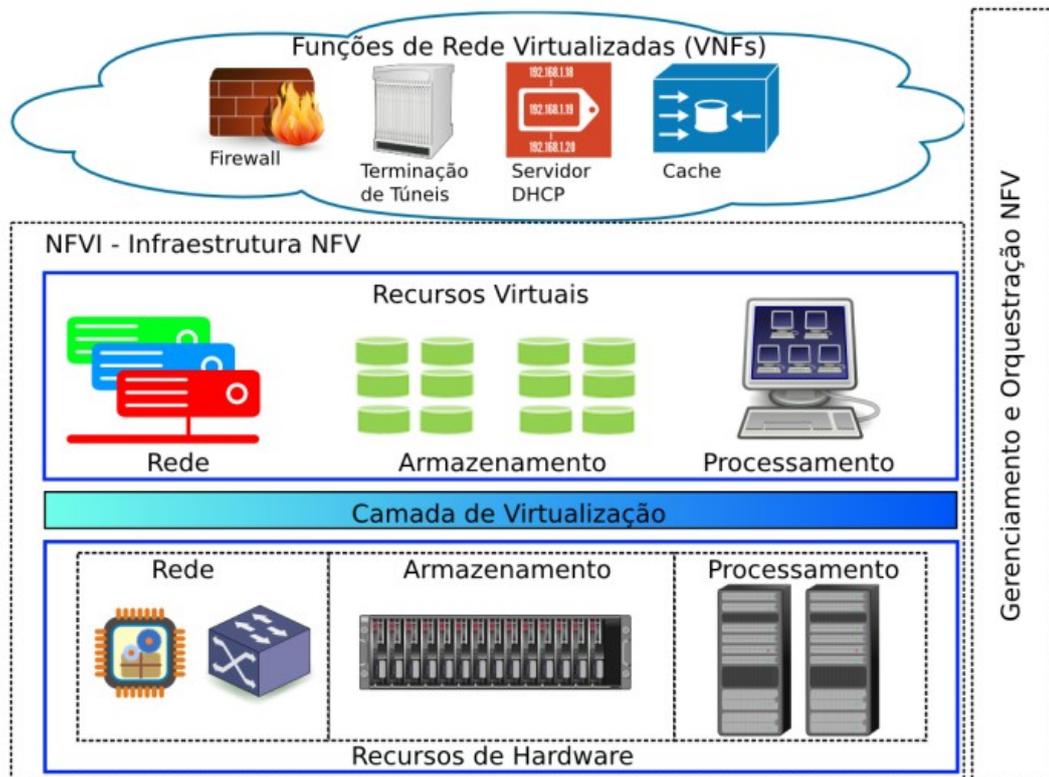
Figura 9

8. Introdução a NFV

A abordagem técnica sobre a NFV (Network Functions Virtualization), pode ser definida como a substituição do hardware dedicado (switch, roteadores, balanceadores de carga, entre outros equipamentos), que possuem alto custo e um software proprietário para fazer sua gestão e administração, com o NFV é possível fazer o que chamamos de "orquestração" de vários dispositivos em apenas um equipamento de hardware robusto e apropriado para esta finalidade.

O NFV é executada mais especificamente utilizando um sistema operacional que coordena dispositivos virtuais, trabalhando em modo "produção" em uma determinada rede ou várias redes de pequeno até grande porte. Podemos referenciar as máquinas virtuais para ter uma melhor compreensão, porém neste caso, o NFV faz a virtualização de equipamentos de rede, vejamos na figura 10, o modo tradicional sem o uso da tecnologia NFV e com o uso da tecnologia NFV.

Figura 10



9. Conclusão

Em outras palavras, é possível que um ISP (Internet Solution Provider), possa construir uma solução SDN/NFV em um ambiente real. ONOS é implantado como um serviço em um cluster de servidores, onde o controlador ONOS é executado em cada servidor.

A simetria de implantação é uma consideração de design importante, pois permite um failover rápido no caso de uma falha do servidor ONOS. O operador de rede pode adicionar servidores de forma incremental, sem interrupção, conforme necessário para capacidade adicional do plano de controle.

Várias instâncias do ONOS podem trabalhar juntas, para criar o que parece para o resto da rede e aplicativos como uma plataforma única. Os aplicativos e dispositivos de rede não precisam saber se estão trabalhando com uma única instância ou com várias instâncias do ONOS. Este recurso torna o ONOS escalonável - pode-se escalar a capacidade do ONOS perfeitamente. É o Distributed Core que faz o trabalho pesado para concretizar esses recursos, esta é a eficiência do controlador ONOS e o futuro das redes SDN aplicado ao negócio e a pesquisa científica.

10. Referências

He, K., Khalid, J., Gember-Jacobson, A., Das, S., Prakash, C., Akella, A., Li, L. E., and Thottan, M. (2015). Measuring control plane latency in sdn-enabled switches. In Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research, page 25. ACM. Jain, R. and Paul, S. (2013). Network virtualization and software defined networking for cloud computing: a survey. Communications Magazine, IEEE, 51(11):24–31. Kim, H. and Feamster, N. (2013). Improving network

management with software defined networking. *Communications Magazine, IEEE*, 51(2):114–119.

Kreutz, D., Ramos, F. M., Esteves Verissimo, P., Esteve Rothenberg, C., Azodolmolky, S., and Uhlig, S. (2015). Software-defined networking: A comprehensive survey. *proceedings of the IEEE*, 103(1):14–76. Lantz, B., Heller, B., and McKeown, N. (2010). A network in a laptop: Rapid prototyping for software-defined networks. In *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks, Hotnets-IX*, pages 19:1–19:6, New York, NY, USA. ACM.

Akyildiz, I. F., Lee, A., Wang, P., Luo, M., and Chou, W. (2014). A roadmap for traffic engineering in sdn-openflow networks. *Computer Networks*, 71:1–30. Benson, T., Akella, A., and Maltz, D. A. (2009). Unraveling the complexity of network

management. In *NSDI*, pages 335–348. Bifulco, R. and Matsiuk, A. (2015). Towards scalable sdn switches: Enabling faster flow table entries installation. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, pages 343–344. ACM. Dusi, M., Bifulco, R., Gringoli, F., and Schneider, F. (2014). Reactive logic in software- defined networking: Measuring flow-table requirements. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2014 International*, pages 340–345. IEEE.