

Introdução

- ❖ A Virtualização de Funções de Rede (*Network Function Virtualization*, NFW) permite transferir funções de rede (como *firewall* e *proxy*), de equipamentos especializados (em geral de alto custo), para soluções em software em servidores de prateleira, usando virtualização. Aliado ao paradigma de Redes Definidas por Software (*Software-Defined Networking*, SDN) é possível que funções virtuais de rede (*Virtual Network Functions*, VNFs) sejam posicionadas de forma flexível na rede.
- ❖ Em trabalho anterior, desenvolveu-se um ambiente cujo objetivo foi assegurar o encaminhamento de fluxos por instâncias de VNFs hospedadas em N-PoPs tal como definido em requisições de encadeamentos de funções de rede (*Service Function Chains*, SFCs).

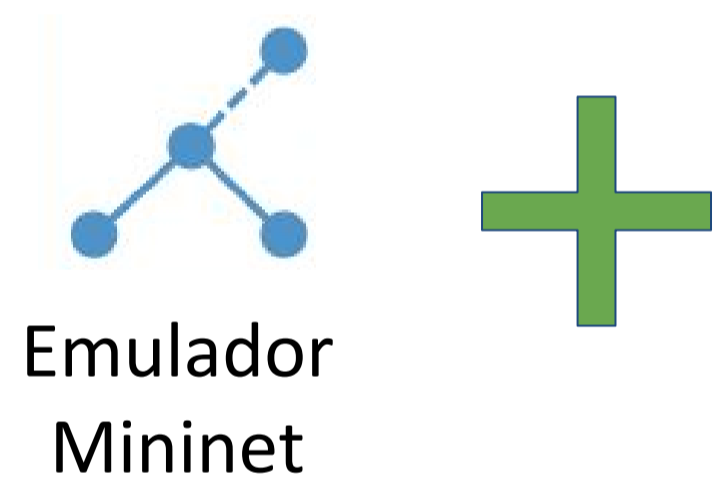
Problemas

- ❖ Fraco acoplamento do ambiente desenvolvido com aquele utilizado para a prototipação e a experimentação de redes SDN/OpenFlow.
- ❖ Complexidade vinculada à falta de praticidade no *setup* de cenários de experimentação.

Objetivos

- ❖ Estender a ferramenta Mininet para facilitar a prototipação e o *setup* de cenários de experimentação utilizando VNFs.
- ❖ Ampliar a oportunidade de uso da proposta de encadeamento de funções.

Proposta



Sistema de instanciação de VNFs e encadeamento de fluxos

Tecnologias

- ❖ *Containers* Docker em *hosts* para conterem as funções de rede
- ❖ Software Click para executar funções de rede programadas em linguagem *click*
- ❖ API desenvolvida pelo Grupo de Redes em trabalho anterior



NFVClickNet

Vantagens

- ❖ Comandos simples para *deploy* e instanciação de VNFs e encadeamento
- ❖ *Setup* prático com diversas possibilidades de uso de VNFs diferentes
- ❖ *Open Source*

Resultados

- ❖ Comandos *deploy* e *enable* implementados de fácil utilização.

```
NFVClickNet> deploy d1 firewall
Deploying function 'firewall'...
Function deployed with success!
NFVClickNet> enable d1 firewall
Enabling function 'firewall'...
Enabled
NFVClickNet>
```

Figura 1: Comandos sendo utilizados no NFVClickNet

- ❖ Execução de VNF com sucesso em *container* Docker.

```
--- 10.0.0.3 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6001ms
rtt min/avg/max/mdev = 0.166/1.268/7.552/2.566 ms
```

Figura 2: Ping realizado passando através de um *firewall* que permite tráfego ICMP

```
NFVClickNet> h3 iperf -s -p 8001 &
NFVClickNet> h1 iperf -c 10.0.0.3 -p 8001
```

Figura 3: Tráfego TCP descartado pelo *firewall* por causa das regras definidas na VNF

- ❖ Comando *sfc* para a geração de regras através de uma SFC.

```
nsbrito@ubuntu-mfv:~$ sudo ovs-ofctl dump-flows s2
NXST_FLOW reply (xid=0x4):
cookie=0x0, duration=255.874s, table=0, n_packets=0, n_bytes=0, idle_age=255, ipv6 actions=drop
cookie=0x0, duration=92.987s, table=0, n_packets=0, n_bytes=0, idle_age=92, priority=1096,ip,in_port=3,
nw_dst=10.0.0.3,nw_tos=0 actions=output:4
cookie=0x0, duration=92.923s, table=0, n_packets=0, n_bytes=0, idle_age=92, priority=1096,ip,in_port=3,
nw_dst=10.0.0.1,nw_tos=0 actions=output:1
cookie=0x0, duration=92.988s, table=0, n_packets=0, n_bytes=0, idle_age=92, priority=1024,ip,nw_dst=10.
0.0.3,nw_tos=0 actions=output:2
cookie=0x0, duration=92.987s, table=0, n_packets=0, n_bytes=0, idle_age=92, priority=1096,ip,nw_dst=10.
0.0.3,nw_tos=0 actions=output:2
cookie=0x0, duration=92.923s, table=0, n_packets=0, n_bytes=0, idle_age=92, priority=1024,ip,nw_dst=10.
0.0.1,nw_tos=0 actions=output:2
cookie=0x0, duration=92.922s, table=0, n_packets=0, n_bytes=0, idle_age=92, priority=1096,ip,nw_dst=10.
0.0.1,nw_tos=0 actions=output:2
cookie=0x0, duration=255.876s, table=0, n_packets=0, n_bytes=0, idle_age=255, priority=0 actions=drop
```

Figura 4: Regras criadas em um *switch* de forma automática através de uma SFC que encaminham para a função de rede especificada

Próximas Etapas

- ❖ Desenvolvimento completo do comando *sfc* - que funciona através de execução manual de diversas partes da API - para a execução através de um *script* automatizado.
- ❖ Melhorias na usabilidade e documentação.
- ❖ Realização de comparativos para verificar a eficiência da solução proposta.
- ❖ O projeto ainda segue em desenvolvimento e está aberto para contribuições de terceiros.