

Uma Plataforma NFV-MANO para Suporte e Orquestração de Serviços de Rede Virtualizados em Nuvem CloudStack

José Flauzino, Elias P. Duarte Jr.

Universidade Federal do Paraná (UFPR)
Curitiba – PR – Brasil

{jwvflauzino,elias}@inf.ufpr.br

Abstract. *Network Function Virtualization (NFV) has the potential to change the way in which network cores are built and managed. The NFV-MANO architecture has been widely used by NFV platforms. These platforms use several enablers - including cloud platforms. In this respect, the predominance of the OpenStack platform is notable. This work proposes Vines, the first CloudStack solution for the orchestration of virtualized network functions and services. The architecture of Vines allows it to holistically and completely manage the full lifecycle of heterogeneous VNFs (Virtual Network Functions) and SFCs (Service Function Chains). Vines was implemented together with CloudStack, and is publicly available. Results of an empirical evaluation are presenting, confirming the feasibility of the platform, also in comparison with OpenStack/Tacker.*

Resumo. *A Virtualização de Funções de Rede (NFV) tem o potencial de revolucionar a forma como os núcleos das redes são construídos e gerenciados. A arquitetura NFV-MANO tem sido amplamente empregada pelas plataformas NFV. Estas plataformas NFV utilizam diversos facilitadores - incluindo nuvens computacionais, com notável predominância do OpenStack. Este trabalho propõe o Vines, a primeira solução CloudStack para a orquestração de funções e serviços de rede virtualizados. Uma contribuição relevante do Vines é uma arquitetura de gerência inovadora e completa de VNFs (Virtualized Network Functions) bem como orquestração de SFCs (Service Function Chains). O Vines foi implementado integrado ao CloudStack e está publicamente disponível. Os resultados de uma avaliação empírica ressaltam a viabilidade da proposta, inclusive ao comparar seu desempenho com o OpenStack/Tacker.*

1. Introdução

A Virtualização de Funções de Rede (NFV - *Network Function Virtualization*) propõe a dissociação entre os equipamentos físicos de rede e as funções de rede (*Network Function* - NF). Isto permite que NFs com funcionalidades diversas como *firewall*, detector de intrusão, entre outras, que tradicionalmente são executadas em hardware especializado, possam ser implementadas como software e executadas como VNFs (*Virtualized Network Functions*). As VNFs são executadas em hardware de propósito geral, através de técnicas de virtualização. Múltiplas VNFs podem ser encadeadas para compor serviços de rede complexos por meio de *Service Function Chaining* (SFC). Através do paradigma NFV é esperado um aumento de flexibilidade, maior facilidade de gerenciamento e a diminuição de custos operacionais (OPEX) e de capital (CAPEX) [Martins et al. 2014].

O *European Telecommunications Standards Institute* (ETSI) vem propondo a arquitetura de referência NFV-MANO (NFV - *Management and Orchestration*) [ETSI 2014], a qual define diversos blocos funcionais. Três blocos particularmente importantes são o *VNF Manager* (VNFM), responsável principalmente pelo gerenciamento do ciclo de vida de VNFs; o *NFV Orchestrator* (NFVO), encarregado, sobretudo, de gerenciar o ciclo de vida de serviços de rede; e o *Virtualized Infrastructure Manager* (VIM), que é incumbido do gerenciamento e a orquestração de recursos da infraestrutura.

Consideradas elementos facilitadores para NFV, as plataformas de computação em nuvem são empregadas como VIM em ambientes NFV. Porém, praticamente todas as atuais plataformas NFV oferecem suporte a um VIM em particular: o OpenStack [OpenStack 2022]. Nesta situação estão plataformas como o OSM (*Open Source MANO*) [ETSI 2022], OpenBaton [OpenBaton 2022] e OPNFV [OPNFV 2022], além do Tacker [Tacker 2022], que é um projeto OpenStack. Por outro lado, o Apache CloudStack [ASF 2022], apesar de sua posição de destaque, sendo uma das plataformas de nuvem de código aberto mais adotadas no mundo [Flexera 2022], tem sido superficialmente explorado no contexto do NFV-MANO.

As plataformas NFV devem permitir tanto o gerenciamento da NF (enquanto software de rede), quanto do ciclo de vida das VNFs (enquanto instâncias virtuais). Neste sentido, mesmo considerando as plataformas NFV baseadas no OpenStack, todas apresentam limitações de gerência das NFs, pois geralmente exigem ações manuais dos operadores de rede. Uma forte motivação para isso, é o fraco suporte a VNF-ExP (*VNF Execution Platform*), que são hospedeiros virtuais especializados para execução de NFs, tais como ClickOS [Martins et al. 2014], Click-on-OSv [Marcuzzo et al. 2017], COVEN [Garcia et al. 2019], entre outras.

Este trabalho propõe o Vines, uma solução NFV de código aberto projetada de acordo com a arquitetura de referência NFV-MANO e disponibilizada como parte da plataforma CloudStack. Uma contribuição relevante do Vines é sua arquitetura de gerência inovadora e completa de VNFs bem como orquestração de SFCs. O núcleo do Vines é constituído por um VNFM e um NFVO, mas conta ainda com uma VNF-ExP denominada Leaf, um EMS (*Element Management System*) abrangente e efetivo, além do suporte a *VNF Packages* (VNFPs). Por conta de sua arquitetura, que explora em profundidade o NFV-MANO, o Vines é capaz de gerenciar de forma holística VNFs heterogêneas e orquestrar serviços de rede implementados como SFCs. Os resultados experimentais (Seção 3) de uma avaliação empírica comprovam a eficácia do Vines ao demonstrar sua capacidade de realizar todas as operações para as quais foi projetado. Sobretudo, os resultados indicam um nível satisfatório de eficiência da proposta, uma vez que seu desempenho é similar ao do OpenStack/Tacker, quando comparados.

2. A Plataforma NFV Cloudstack/Vines

Uma visão geral da relação entre o CloudStack/Vines e a arquitetura NFV-MANO é apresentada na Figura 1. Os módulos tradicionais do CloudStack (a própria plataforma de nuvem) podem ser compreendidos como o bloco funcional VIM do NFV-MANO. Assim, o Vines estende as capacidades do CloudStack, que passa a ofertar as funcionalidades de diversos blocos do NFV-MANO, abstraindo detalhes de virtualização e provendo uma interface de alto nível para o gerenciamento e orquestração de funções e serviços de rede.

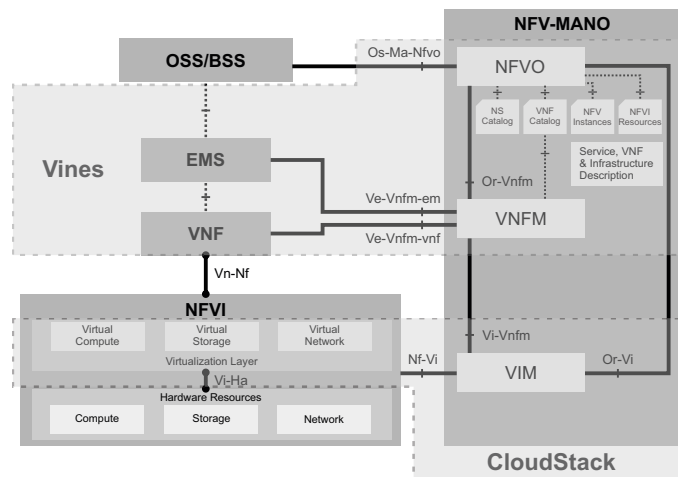
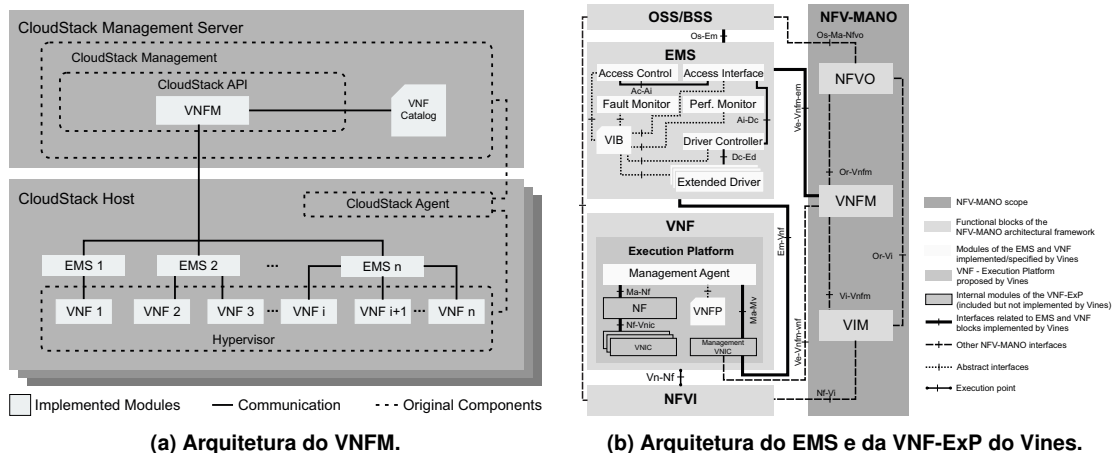


Figura 1. O CloudStack/Vines em relação ao NFV-MANO.

Esta seção está organizada em duas subseções. A Subseção 2.1 descreve a arquitetura de gerenciamento de VNFs, enquanto a Subseção 2.2 apresenta a arquitetura de orquestração de serviços de rede.

2.1. Gerência do Ciclo de Vida de Funções de Rede Virtualizadas

O VNFM do Vines, através do relacionamento com outros componentes da arquitetura, disponibiliza um conjunto completo de funcionalidades de gerenciamento do ciclo de vida de VNFs heterogêneas, que podem ser compostas por NFs com diferentes implementações e instanciadas a partir de VNF-ExPs distintas. A Figura 2(a) apresenta o VNFM e os demais módulos da arquitetura relacionados à gerência de VNFs. O Vines mantém diversos repositórios NFV. Um deles é o *VNF Catalog*, composto por um conjunto de VNFPs (*VNF Packages*) baseados na especificação *TOSCA YAML Cloud Service Archive (CSAR)* [ETSI 2018]. Os VNFPs são fundamentais para o gerenciamento das VNFs, pois armazenam os arquivos cruciais para cada operação (e.g., *scripts* de gerenciamento da NF e imagens de software).



(a) Arquitetura do VNFM.

(b) Arquitetura do EMS e da VNF-Exp do Vines.

Figura 2. Arquitetura de gerência de VNF do Vines.

A comunicação entre as aplicações *CloudStack Management* e *CloudStack Agent*, é a base para o VNFM do Vines realizar a gerência do ciclo de vida de VNFs (e.g.,

a instanciação, escala de recursos ou remoção das VNF-ExPs). Mas o Vines inclui ainda uma estratégia para o gerenciamento completo de VNFs, que é realizada através da comunicação entre o VNFM e cada EMS das respectivas VNFs. Para isto, é inclusa a definição detalhada da estrutura de um EMS flexível o suficiente para lidar com VNFs heterogêneas de forma efetiva. O EMS projetado, detalhado na Figura 2(b), leva em consideração as interfaces especificadas pelo ponto de referência *Ve-Vnfm-em* do NFV-MANO. Porém, uma vez que o NFV-MANO menciona a necessidade de comunicação tanto entre EMS e VNF, quanto OSS/BSS (*Operations Support System/Business Support System*) e EMS, mas não especifica as interfaces envolvidas, o Vines preenche esta lacuna nomeando estes pontos de referência e definindo suas interfaces.

Assim, neste trabalho o ponto de referência entre OSS/BSS e EMS é nomeado de *Os-Em*, enquanto *Em-Vnf* representa a ligação entre EMS e VNF. As interfaces são especificadas para estes pontos de referência a partir daquelas previamente definidas pelo NFV-MANO para os pontos *Ve-Vnfm-em* e *Ve-Vnfm-vnf* [ETSI 2021]. No *Os-Em* são empregadas as mesmas interfaces do *Ve-Vnfm-em*. Já o ponto de referência *Em-Vnf* inclui as operações do *Ve-Vnfm-vnf*, com exceção da interface VNF *Indicator*, pois o EMS não se inscreve na VNF para receber notificações – ao contrário, o EMS monitora ativamente a VNF e provê informações da VNF a partir de seu próprio serviço de notificação.

No EMS, o módulo *Access Interface* expõe interfaces de acesso ao conjunto de operações do EMS (especificadas pelo *Ve-Vnfm-em* do NFV-MANO e pelo *Os-Em*, especificado neste trabalho). Já o *Access Control* é responsável por autenticar cada requisição recebida pelo EMS. A VNF *Information Base* (VIB) mantém as informações providas pelo VNFM a respeito das VNFs que são de responsabilidade do EMS. Os módulos *Fault Monitor* e *Performance Monitor* monitoram as VNFs e notificam o VNFM com base em políticas de monitoramento de falhas e de uso de recursos. Cada requisição recebida pelo módulo *Access Interface* é encaminhada pelo Driver Controller para um *Extended Driver* (definido a seguir) através do qual se comunica com a VNF e trata devidamente das especificidades da VNF-ExP. Por fim, o *Extended Driver* atua como intermediário da comunicação entre EMS e VNF, podendo realizar funcionalidades como a conversão de parâmetros, remontagem de URIs, executar diferentes métodos de chamada (HTTP, SSH, *socket*, etc.), entre outras. Deve haver um *Extended Driver* para cada tipo de VNF-ExP.

Visando oferecer um ambiente flexível para a execução de NFs, o Vines inclui a definição de uma arquitetura para VNF-ExP - detalhada na Figura 2(b), internamente ao bloco VNF. Sua estrutura minimalista e versátil, possibilita a implementação de VNF-ExPs simplificadas e com poucas restrições quanto aos tipos de NFs que podem ser executadas. Uma VNF-ExP desenvolvida com base nesta arquitetura permite a execução de NFs legadas ou quaisquer outras NFs, inclusive as que estiverem em desenvolvimento.

2.2. Orquestração de Serviços de Rede Virtualizados

O NFVO do Vines é projetado para ser capaz de compor e orquestrar serviços de rede complexos sobre as redes virtuais nativas do CloudStack. No CloudStack, as VMs de clientes são instanciadas em redes do tipo *Guest*. Cada rede *Guest* é isolada e a comunicação de uma VM com outras redes *Guest* ou redes externas à nuvem é roteada através de uma *System VM* chamada VR (*Virtual Router*), que atua como o *gateway* da rede. É neste tipo de rede que as VNFs são instanciadas pelo Vines. Por funcionarem como redes locais,

quando VNFs pertencentes à mesma rede *Guest* se comunicam entre si, o VR da rede atua como um *switch* virtual (apenas comutando os pacotes de rede). Em outras palavras, quaisquer comunicações deste tipo não passam pelas regras de roteamento do VR (*i.e.*, nenhuma rota definida surte efeito nesses tráfegos).

Cada VNIC (*Virtual Network Interface Card*) de uma instância de VNF possui um IP privado, usado internamente à respectiva rede *Guest* em que está conectada. Para possibilitar a comunicação das VNFs intra-redes, o VR associa um IP da rede pública do CloudStack a determinado IP privado de uma VNF. Essa associação é feita internamente ao VR, sem qualquer relação com a configuração da VNIC da VNF. Assim, através de SNAT (*Source Network Address Translation*), o VR provê a uma VNF o acesso a outras redes *Guest* ou externas à nuvem (*e.g.*, a Internet).

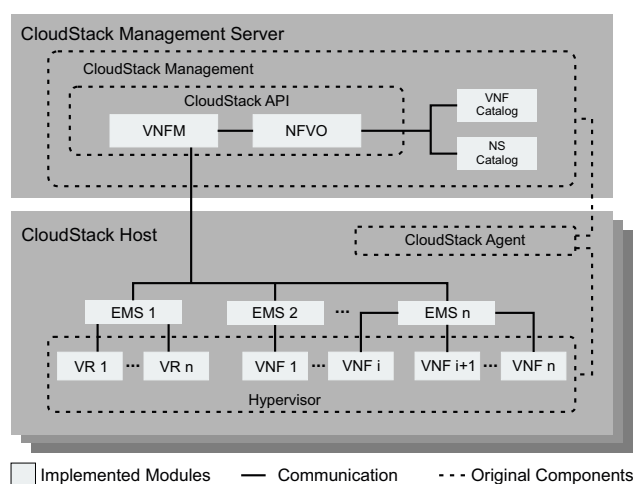


Figura 3. Arquitetura do NFVO.

A Figura 3 apresenta uma visão geral da arquitetura de orquestração de serviços do Vines. Nela é ilustrado o relacionamento do NFVO com os demais componentes do Vines. O NFVO do Vines está contido na aplicação principal do CloudStack (*CloudStack Management*). O Vines também mantém um repositório chamado *Network Service Catalog*. Este repositório é composto por descritores como NSD (*Network Service Descriptor*) e VNFFGD (*VNF Forward Graph Descriptor*), que especificam detalhes do encadeamento de VNFs para compor cada serviço de rede. A arquitetura do NFVO prevê seu relacionamento com todos os blocos funcionais do NFV-MANO que são fundamentais para suas atividades de orquestração. Assim, o NFVO possui acesso direto aos repositórios NFV e ao VNFM. Já o interfaceamento com o VIM (o próprio CloudStack) dá-se através das aplicações *CloudStack Management* e *CloudStack Agent*.

Visando alcançar maior centralização do plano de controle das redes virtuais do CloudStack, a estratégia adotada foi permitir que o NFVO utilize toda a capacidade de gerenciamento do VNFM. O ponto chave desta estratégia é tratar cada VR do CloudStack como uma VNF interna ao sistema – em contraposição com VNFs de usuário. Desta forma, além de facilitar a orquestração, os próprios recursos de rede do CloudStack são tratados da mesma forma que qualquer VNF, inclusive tendo um EMS responsável por seu gerenciamento. A Figura 3 demonstra que cada EMS pode ser responsável por uma ou mais VNFs e também por um ou mais VRs. Ao empregar a arquitetura de gerência

de VNFs do Vines como apoio à orquestração de rede, o NFVO do Vines herda todas as características daquela arquitetura. Esta abordagem concede ao NFVO capacidade de gerência suficiente para realizar as configurações necessárias nos elementos de rede ao compor os serviços de rede virtualizados.

Cada serviço de rede é criado pelo Vines através da composição de SFCs. Considerando as especificidades das redes virtuais nativas do CloudStack (mencionadas anteriormente), a estratégia do NFVO do Vines para a composição de SFCs envolve configurar tanto o VR que atua como *gateway* na rede *Guest* das VNFs que irão compor a SFC, quanto as próprias VNFs. O direcionamento de tráfego para uma SFC construída pelo Vines é realizado através do encaminhamento do tráfego para um endereço IP pertencente à rede pública da nuvem CloudStack. Ou seja, cada SFC possui um IP da rede de tráfego público do CloudStack associado a ela, utilizado pelos clientes ao acessarem as funcionalidades providas pelo serviço. Concretamente, ao compor uma SFC, o NFVO do Vines atribui um endereço IP público para a última VNF e associa este endereço à SFC. Em seguida, configura o VR de modo que o tráfego com direção ao IP público da SFC (*i.e.*, direcionado à última VNF) seja desviado para a primeira VNF da SFC. O NFVO configura as primeiras $n-1$ VNFs da SFC (sendo n o número total de VNFs) para encaminharem o fluxo para a próxima VNF, até chegar à última VNF. Por ser o destino “real” da SFC ela processa os dados e depois encaminha o resultado ao remetente através do VR.

A arquitetura proposta foi implementada (com cerca de 8.000 linhas de código) e o Vines está publicamente disponível¹. O núcleo do Vines (VNFM e NFVO) foi construído junto ao código fonte do próprio CloudStack. Ainda com base na arquitetura do Vines, foram implementados um EMS e a VNF-Exp denominada Leaf (ambos são ofertados como *templates* de VMs na plataforma). As funcionalidades de gerência e orquestração disponibilizadas pelo Vines são acessadas via API do CloudStack.

3. Resultados

Esta seção apresenta os resultados de uma avaliação empírica utilizando um servidor com processador Intel Core i7@3.40GHz, 8G de RAM e uma NIC 1Gb Ethernet. As cargas de trabalho foram geradas a partir de uma máquina cliente conectada diretamente ao servidor.

Um primeiro experimento visa avaliar o tempo médio de execução das operações de gerência do ciclo de vida de VNFs com diferentes VNF-ExPs. Uma vez que o OpenStack/Tacker não trata das especificidades de VNF-ExPs, apenas o CloudStack/Vines é avaliado neste experimento. A Figura 4 apresenta os resultados obtidos (com um intervalo de confiança de 95%) ao gerenciar as VNF-ExPs Vines-Leaf, Click-on-OSv e COVEN. O CloudStack/Vines foi capaz de realizar todas as operações em menos de 60 segundos, sendo a grande maioria realizadas em um tempo médio inferior a 40 segundos. Observa-se ainda que o tempo para executar as operações varia de acordo com a VNF-Exp utilizada.

As figuras 5(a) e 5(b) apresentam, respectivamente, a distribuição da latência de SFCs com variadas quantidades de VNFs encadeadas instanciadas nas plataformas CloudStack/Vines e OpenStack/Tacker. As latências de ambas as plataformas foram bastante similares. A maior diferença do pico de concentração da latência entre SFCs das duas plataformas - cenário com 5 VNFs da Figura 5(b) - foi de cerca de 0,0009 segundos. Nota-se ainda uma dispersão da latência em SFCs de 4 e 5 VNFs no OpenStack/Tacker.

¹<http://www.inf.ufpr.br/jwvflauzino/vines>

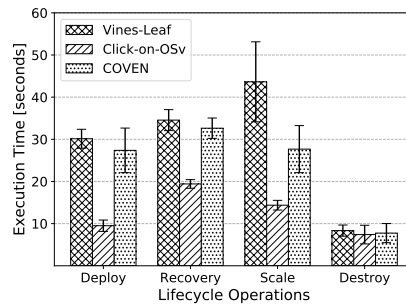


Figura 4. Tempo médio de execução das operações de gerência do ciclo de vida.

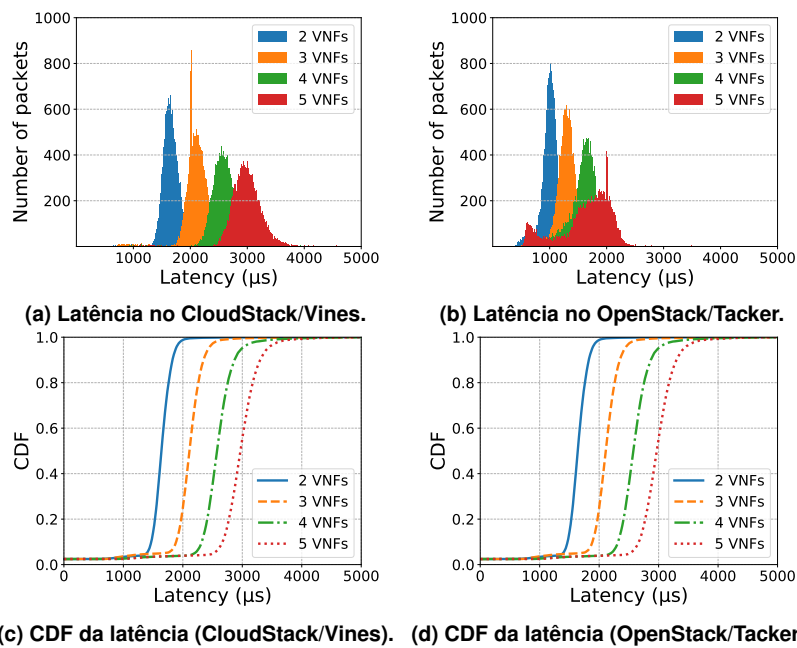


Figura 5. Latência das SFCs no CloudStack/Vines e no OpenStack/Tacker.

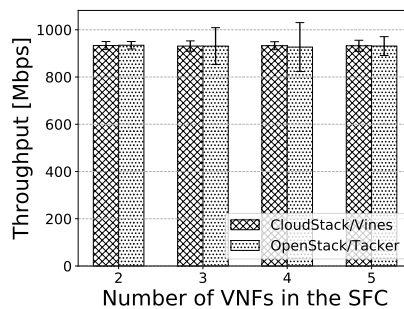


Figura 6. Vazão das SFCs entre o CloudStack/Vines e o OpenStack/Tacker.

Com base na latência dos cenários apresentados nas figuras 5(a) e 5(b), as figuras 5(c) e 5(d) evidenciam a probabilidade de um tráfego de rede sofrer determinada latência ao ser encaminhado para cada uma das SFCs. Estes resultados indicam que há cerca de 95% de chance de uma carga de trabalho obter uma latência inferior ou igual a $1890\mu s$, $2430\mu s$, $2990\mu s$ e $3430\mu s$, respectivamente, para SFCs com 2, 3, 4 e 5 VNFs instanciadas pelo CloudStack/Vines. No OpenStack/Tacker, a probabilidade é de 95% da latência não

exceder $1150\mu\text{s}$, $1500\mu\text{s}$, $1860\mu\text{s}$ e $2190\mu\text{s}$ em cada um dos respectivos cenários.

Por fim, a Figura 6 apresenta a vazão das SFCs de ambas as plataformas NFV nos mesmos cenários. Em todos os cenários, nas duas plataformas, a taxa de transferência média foi de ~ 930 Mbps. Os resultados demonstram uma grande similaridade de ambas as plataformas, sendo notável uma maior variação da vazão das SFCs do OpenStack/Tacker.

4. Publicações

Os resultados dessa dissertação geraram as seguintes publicações:

- Flauzino, J. et al. “Além do OpenStack: Disponibilizando o Suporte para Funções Virtualizadas de Rede NFV-MANO no CloudStack”. SBRC, 2020.
- Flauzino, J. et al. “Gerência e Orquestração de Funções e Serviços de Rede Virtualizados em Nuvem CloudStack”. WGRS, 2021.
- Flauzino, J. et. al. “Vines: An NFV-MANO Platform for the Management and Orchestration of Virtualized Network Functions and Services on CloudStack-based Clouds”, submetido ao periódico *Software Practice and Experience*.

5. Conclusão

Este trabalho propõe o Vines, a primeira solução NFV para o Apache CloudStack. A principal contribuição é a arquitetura holística de gerência e orquestração de serviços de rede virtualizados. Resultados experimentais comprovam a eficácia do Vines, com desempenho similar ao do OpenStack/Tacker. Trabalhos futuros incluem adicionar novas funcionalidades, como a reclassificação dinâmica de tráfego (*branching*), bem como encaqueamento de VNFs baseado em NSH (*Network Service Header*).

Referências

- ASF (2022). Apache CloudStack. <http://cloudstack.apache.org>.
- ETSI (2014). NFV: Management and Orchestration. Technical report, ETSI.
- ETSI (2018). NFV Rel. 2; Protocols Data Models; VNFP spec. Technical report, ETSI.
- ETSI (2021). NFV Release 3; Management and Orchestration; Ve-Vnfm ref. point - Interface and Information Model Specification. Technical report, ETSI.
- ETSI (2022). Open Source MANO. <https://osm.etsi.org>.
- Flexera (2022). State of the Cloud Report. www.flexera.com.
- Garcia, V. F. et al. (2019). An NSH-enabled architecture for Virtualized Network Function platforms. In *IEEE AINA*, pages 376–387. Springer.
- Marcuzzo, L. d. C. et al. (2017). Click-on-OSv: A platform for running click-based middleboxes. In *IEEE IFIP IM*, pages 885–886. IEEE.
- Martins, J. et al. (2014). ClickOS and the art of network function virtualization. In *Symposium on Networked Systems Design and Implementation*, pages 459–473. USENIX.
- OpenBaton (2022). Open Baton. <https://openbaton.github.io>.
- OpenStack (2022). Main Page. <https://wiki.openstack.org/wiki>.
- OPNFV (2022). Open Platform for NFV. <https://www.opnfv.org>.
- Tacker (2022). Tacker/OpenStack. wiki.openstack.org/wiki/Tacker.