Uma Arquitetura para Gerenciamento e Pesquisa Experimental de Funções Virtualizadas para Computação em Borda

Hugo Gustavo Valin Oliveira da Cunha[1], **Rodrigo Moreira**[1,2], **Flávio de Oliveira Silva**[1]

[1] Faculdade de Computação (FACOM) – Universidade Federal de Uberlândia (UFU)
38.408-100 – Uberlândia – MG – Brasil

[2]Instituto de Ciências Exatas e Tecnológicas (IEP) – Universidade Federal de Viçosa (UFV)
38.810-000 – Rio Paranaíba – MG – Brasil.

{hugo.cunha, rodrigo.moreira, flavio}@ufu.br, rodrigo@ufv.br

Abstract. *Edge Computing was proposed to solve problems such as bandwidth management, geographic dispersion, allowing data processing close to end users. The goal of Network Functions Virtualization (NFV) is to simplify the workflow for creating computational services. This work advances the state of the art by proposing a unified management architecture of Virtualized Everything Functions (VxFs) in Edge Computing. This architecture is capable of continuously deploying and monitoring VxFs regardless of hardware and virtualization platform. The proposed architecture contributes to the realization of the most diverse experimental research scenarios on the edge, contributing to advance knowledge in this area.*

Resumo. Edge Computing *foi proposto para resolver problemas como gerenciamento de largura de banda, dispersão geográfica, permitindo o processamento de dados próximo aos usuários finais. O objetivo de Network Functions Virtualization (NFV) é simplificar o fluxo de trabalho para criação de serviços computacionais. Este trabalho avança o estado da arte ao propor uma arquitetura gerenciamento unificado de* Virtualized Everything Functions *(VxFs) em* Edge Computing. *Essa arquitetura é capaz de implantar e monitorar continuamente VxFs, independentemente do hardware e a plataforma virtualização. A arquitetura proposta contribui para a realização dos mais diversos cenários de pesquisa experimental na borda, contribuindo para avançar o conhecimento nesta área.*

1. Introdução

A virtualização é um dos habilitadores tecnológicos centrais para a construção de aplicativos de IoT versáteis. Tal tecnologia tem a natureza de compartilhar e isolar recursos computacionais para diferentes usuários, proporcionando novos paradigmas de computação como *Edge Computing*. *Edge Computing* propõe a transferência de parte do processamento centralizado de tarefas em *Cloud Computing* para a borda da rede, resolvendo problemas como gerenciamento de largura de banda, dispersão geográfica, baixa latência e problemas de privacidade.

Desta forma, este trabalho avança o estado da arte ao propor e avaliar uma arquitetura experimental compatível com ETSI *Management and Orchestration Platform*

(MANO) para *Virtualized Everything Functions*(VxFs). Esta arquitetura é capaz de implantar e monitorar continuamente VxFs, utilizando ferramentas comuns de livre acesso, para os mais diferentes experimentos, independentemente do hardware e a plataforma virtualização. Com estas características trata-se de um ambiente que permite realizar os mais diversos experimentos com possibilidades de coletar métricas das diversas camadas envolvidas neste tipo de aplicação.

Este trabalho está organizado da seguinte forma: a Seção 2 apresenta trabalhos relacionados. A Seção 3 apresenta a arquitetura para pesquisa experimental com funções de rede na borda. A Seção 4 apresenta diversas métricas e capacidades para obtenção de dados experimentais. A Seção 5 apresenta comentários finais.

2. Trabalhos Relacionados

Uma proposta experimental em *Edge Computing* que leva o serviço de *streaming* de vídeo mais perto do cliente para economizar largura de banda, energia e outros custos, é descrita em [Rigazzi et al. 2019]. A abordagem dos autores avançou o entendimento dos orquestradores para domínio de borda ao apresentar um conceito de orquestração de serviços dentro do ecossistema 5G-CORAL, que unifica os recursos MEC e NFV de forma federada,

Em [Baldoni et al. 2018] os autores apontam lacunas e posteriormente descrevem uma proposta de alto nível para unificar o ciclo de funções virtualizadas nos domínios MEC e NFV através de uma camada sobre esses orquestradores. Essa camada considera a existência de análises adequadas para cada orquestrador, capazes de separar e analisar o arquivo descritor de serviço e encaminhá-lo ao orquestrador apropriado.

O projeto 5GCity [Paolino et al. 2019] descreve uma proposta de plataforma com uma máquina hospedeira neutra oferecendo divisão de recursos, como computação, rede e armazenamento ponta a ponta, considerando os requisitos de segurança. O orquestrador proposto representa o *Virtual Infrastructure Manager* (VIM) específico para cada domínio tecnológico, oferecendo extensões de autenticação, monitoramento do sistema, ativos e *geotagging*.

Os autores [Tambe et al. 2020] realizaram uma avaliação experimental, considerando o OSM como orquestrador do serviço, e considerados KPIs de tempo de instanciação e tempo de resposta do serviço implantados em máquinas virtuais convencionais e para contêineres na borda da rede.

Uma lacuna comum a todos estes trabalhos é que não suportam o monitoramento multi-camadas de toda a infraestrutura física e de software. Outra lacuna comum a todos é a dependência da arquitetura de hardware seja ela x86 ou ARM.

3. Arquitetura de Gerenciamento Unificado

Esta seção apresenta os componentes da arquitetura de gerenciamento unificado, suas capacidades de monitoramento multi-camadas, que permite coletar as mais diversas métricas em um cenário de pesquisa experimental. A arquitetura avança o estado da arte ao permitir um monitoramento multi-camada completo de toda a infraestrutura e uma independência do hardware utilizado para virtualização.

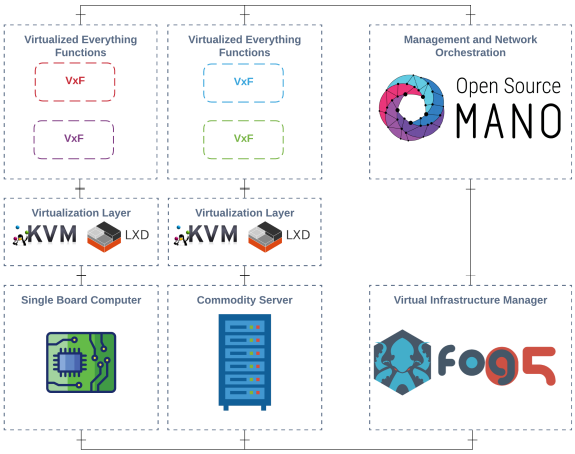

Figura 1. Componentes da arquitetura unificada.

3.1. Arquitetura Geral

A Figura 1 mostra a arquitetura unificada geral do NFV em *Edge Computing* (EC), destacando seus componentes lógicos. Open Source MANO (OSM) é uma pilha de código aberto hospedada por ETSI para NFV, alinhada com *ETSI NFV Information Models*, capaz de consumir descritores de serviços significativos e independente de gerenciador de infraestrutura. Eclipse Fog05 é um projeto de código aberto que permite o gerenciamento ponta a ponta de computação, armazenamento, rede e dispositivos de entrada/saída em *Edge Computing*.

Um computador de placa única, é um computador funcional completo baseado na arquitetura ARM. Um servidor *bare-metal* é um computador dedicado a executar programas baseado em arquitetura x86. Virtualização é uma tecnologia utilizada para executar vários sistemas operacionais ou aplicativos em uma única infraestrutura física, fornecendo a cada um deles uma visão abstrata do *hardware*. Para o presente projeto, duas abordagens de virtualização serão utilizadas, a saber, LXD e KVM. *Virtualized Everything Functions* [Silva et al. 2019] são uma ou mais máquinas virtuais (ou contêineres) que implementam *softwares* especializados ao invés de serem implementados em dispositivos físicos.

3.2. Monitoramento Multi-Camadas

Compreender o estado de um sistema é essencial para garantir a confiabilidade, integridade e desempenho de aplicativos e serviços. A Figura 2 mostra todos os componentes necessários para monitorar e coletar informações sobre o comportamento da infraestrutura, ambiente virtualizado e futuras aplicações que possam ser implantadas. *Network Function Virtualized Infrastructure* (NFVI) é representado pelo *Raspberry Pi 4* ou por um servidor *bare-metal*. *Exporter* é um componente de *software* capaz de expor métricas no formato *OpenMetrics*. *Prometheus* é um *software* capaz de registrar qualquer série temporal puramente numérica. *VictoriaMetrics* é banco de dados para séries temporais. O *Prometheus* foi configurado para armazenar todos os dados coletados no *VictoriaMetrics*. Todos os dados coletados pela NFVI podem ser integrados, analisados e visualizados por *software* de terceiros.

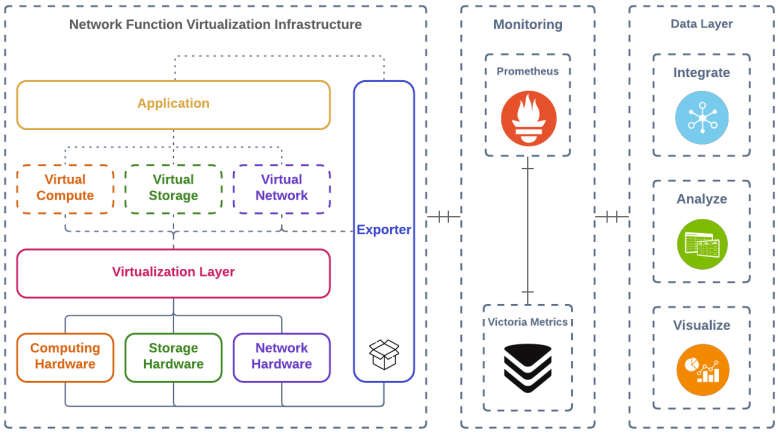

Figura 2. Componentes de monitoramento da arquitetura unificada.

Existem diversas métricas que a arquitetura unificada pode oferecer, como CPU, memória, espaço em disco, largura de banda, tais métricas são podem estar em nível de infraestrutura de nível de ambiente virtualizado, bem como em nível de aplicação. A Figura 3 mostra o diagrama de sequencia de monitoramento da arquitetura. Dado uma infraestrutura de implantação, independente de seu tipo, o monitoramento é capaz de extrair informações de cada uma delas, por meio de requisições HTTP.

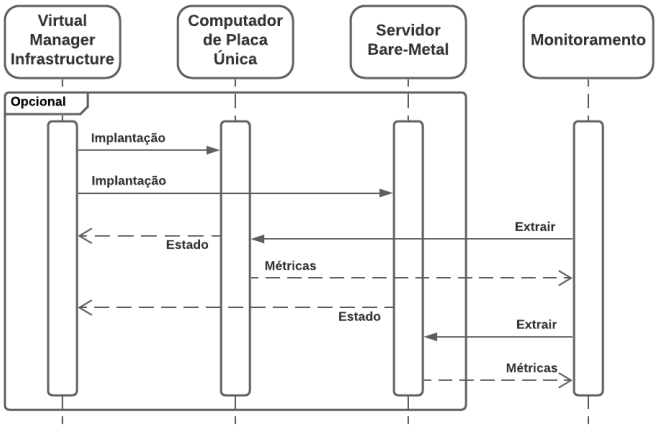

Figura 3. Diagrama de sequência para a arquitetura unificada de monitoramento.

Um exportador é um pedaço de software que você implanta ao lado do aplicativo do qual deseja obter métricas. Existem várias bibliotecas e servidores que ajudam na exportação de métricas existentes de sistemas de terceiros como métricas *Prometheus*. Isso é útil para casos em que não é viável implementar em sistemas específicos um componente de software para monitoramento, como aplicações proprietárias.

Os exportadores mantidos como oficiais no mercado ajudam a obter métricas com informações do sistema, como informações de infraestrutura e ambientes virtualizados. Mas caso se deseja um conjunto específico de métricas que não são expostas a nenhum exportador no mercado, pode considerar o desenvolvimento do seu próprio exportador.

Isso permitirá obter várias métricas de uma só vez, capazes de ser consumidas pela arquitetura unificada, e disponibilizadas de forma transparente e indistinguível a métricas pré-existentes.

A arquitetura unificada permite monitorar todo ambiente de implantação da VxF. Logo, métricas em nível de aplicação, podem ser relacionadas a métricas em nível de infraestrutura, bem como métricas em nível ambiente virtualizado. O monitoramento completo da pilha pode ser parte essencial de um experimento em específico. Deste modo, a arquitetura unificada cria as bases para que novos experimentos, possam ser explorados e monitorados de forma simples e padronizados.

4. Exportadores e métricas

Para mostrar a capacidade de implantar e monitorar continuamente VxFs, utilizando ferramentas comuns de livre acesso, essa seção apresenta alguns exportadores e suas possíveis das métricas.

O *Node Exporter* é exportador de métricas no nível da máquina, em grande parte do kernel do seu sistema operacional, como CPU, memória, espaço em disco, E/S de disco, rede largura de banda e temperatura da placa-mãe. O *Node Exporter* é usado com sistemas Unix. O *Node Exporter* destina-se apenas a monitorar a própria máquina, não os processos ou serviços individuais nela. Eis uma lista de algumas das várias métricas que o *Node Exporter* é capaz de disponibilizar bem como sua definição:

- **node_cpu_seconds_total**: Segundos de cpu gasto em cada modo.
- **node_memory_Buffers_bytes**: Memória total em bytes usados em buffers.
- **node_filesystem_avail_bytes**: Espaço do sistema de arquivos disponível para usuários não root em bytes.
- **node_disk_io_time_seconds_total**: Total de segundos gastos fazendo operações de entrada e saída .
- **node_network_receive_bytes_total**: O número total de bytes recebidos pela rede.

Da mesma forma que o Node Exporter fornece métricas sobre a máquina, o cAdvisor é um exportador que fornece métricas sobre contêineres no Linux. Eis uma lista de algumas das várias métricas que o cAdvisor é capaz de disponibilizar bem como sua definição:

- **container_cpu_schedstat_run_periods_total**: Número de vezes que os processos em contêineres foram executados na cpu.
- **container_fs_io_current**: Número de operações de entrada e saída atualmente em andamento.
- **container_memory_cache**: Memória total do cache da página.
- **container_network_transmit_packets_dropped_total**: Contagem acumulada de pacotes caiu durante a transmissão.

Nem todos os aplicativos produzem métricas em um formato que pode ser convertido em métrica que podem ser consumidas pela arquitetura unificada. Mas tais aplicativos podem produzir logs, e o exportador *Grok* pode ser usado para convertê-los em métricas. O exportador *Grok* reutiliza a mesma linguagem de padrão, permitindo que você reutilize padrões que você já possui.

Outro exportador possível de utilização é *Blackbox Exporter*, que analisa *endpoints* em protocolos HTTP, HTTPS, DNS, TCP ou ICMP, retornando métricas detalhadas sobre a solicitação, incluindo se ela foi bem-sucedida ou não e quanto tempo levou para receber uma resposta. O estudo apresentado em [da Cunha et al. 2021] foi realizado completamente com a arquitetura proposta e mostra sua versatilidade para a realização de diversas métricas em um cenário de pesquisa experimental.

5. Conclusão

Este trabalho propõe uma arquitetura de experimentação e monitoramento compatível com ETSI MANO para o gerenciamento de VxFs. Tal arquitetura é independente do *hardware* de virtualização subjacente para implantação e monitoramento, isto é, seja em computador de placa única a servidores *bare-metal*. A arquitetura unificada pode fornecer uma infraestrutura de monitoramento completa capaz de coletar uma variedade de métricas relacionadas ao hardware de suporte, bem como métricas relacionadas ao ambientes e serviços virtualizados.

A arquitetura unificada proposta permite realizar pesquisa experimental em diversos cenários, contribuindo para avançar o conhecimento nesta área que é crucial para as novas aplicações e serviços em diversos verticais de negócio.

Referências

Baldoni, G., Cruschelli, P., Paolino, M., Meixner, C. C., Albanese, A., Papageorgiou, A., Khalili, H., Siddiqui, S., and Simeonidou, D. (2018). Edge computing enhancements in an nfv-based ecosystem for 5g neutral hosts. In *2018 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, pages 1–5.

da Cunha, H. G. V. O., Moreira, R., and de Oliveira Silva, F. (2021). A Comparative Study Between Containerization and Full-Virtualization of Virtualized Everything Functions in Edge Computing. In Barolli, L., Woungang, I., and Enokido, T., editors, *Advanced Information Networking and Applications*, Lecture Notes in Networks and Systems, pages 771–782, Cham. Springer International Publishing.

Paolino, M., Carrozzo, G., Betzler, A., Colman-Meixner, C., Khalili, H., Siddiqui, S., Sechkova, T., and Simeonidou, D. (2019). Compute and network virtualization at the edge for 5g smart cities neutral host infrastructures. In *2019 IEEE 2nd 5G World Forum (5GWF)*, pages 560–565.

Rigazzi, G., Kainulainen, J. P., Turyagyenda, C., Mourad, A., and Ahn, J. (2019). An edge and fog computing platform for effective deployment of 360 video applications. In *2019 IEEE Wireless Communications and Networking Conference Workshop (WCNCW)*, pages 1–6.

Silva, A. P., Tranoris, C., Denazis, S., Sargento, S., Pereira, J., Luís, M., Moreira, R., Silva, F., Vidal, I., Nogales, B., Nejabati, R., and Simeonidou, D. (2019). 5ginfire: An end-to-end open5g vertical network function ecosystem. volume 93, page 101895.

Tambe, S. D., Mandge, Y., and Antony Franklin, A. (2020). Performance study of multi-access edge computing deployment in a virtualized environment. In *2020 IEEE 3rd 5G World Forum (5GWF)*, pages 424–429.