Mobile Edge Vertical Applications Using ETSI MEC APIs and Sandbox

Rasoul Nikbakht*, Michail Dalgitsis†, Sergio Barrachina-Muñoz*, Sarang Kahvazadeh*
*Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), Castelldefels, Spain
†Vicomtech, San Sebastian, Spain
nikbakht@cttc.es, mdalgitsis@vicomtech.org, sbarrachina@cttc.es, skahvazadeh@cttc.es

Abstract—MEC Sandbox is an excellent tool that simulates wireless networks and deploys ETSI Multi-access Edge Computing (MEC) APIs on top of the simulated wireless network. In this demo, we consume these APIs using a decision engine (DE) to scale a video-on-demand (VoD) application located on the network edge. Specifically, the developed DE uses the ETSI MEC Location API and retrieves the number of users in a given zone. The DE then takes actions at the microservice scaling level and executes them through a custom-made Kubernetes-based OpenAPI.

Index Terms—MEC Sandbox, MEC API, Microservice scaling

I. INTRODUCTION

Within the broad topic of edge computing, Multi-access Edge Computing (MEC) is the widely accepted standard for edge computing created by the European Telecommunications Standards Institute (ETSI). MEC provides computing and storage resources at the edge of 5G networks to let application and mobile operators reduce delay and backhaul traffic. In this work, we leverage the MEC Sandbox assuming simulated users that stream a cloud-native video running at the MEC site, by calling the MEC Location API to calculate the average number of users in a given zone. Should the traffic demand increase, the decision engine (DE) scales the video server and exports metrics to the monitoring system.

II. SYSTEM ARCHITECTURE AND EXPERIMENT

In this demonstration, we showcase the complete pipeline of an edge microservice scaling solution using MEC Sandbox and a custom-made Kubernetes OpenAPI. First, we deploy a Kubernetes cluster consisting of cloud and edge nodes. Then, we use a custom Helm package [1], [2] to deploy a cloud-native 5G network realized with Open5GS network functions that span cloud and edge domains. In addition, we deploy (i) an ingress controller in the Kubernetes edge node for distributing the traffic between the pods of the video-on-demand (VoD) application [3], (ii) the custom-made OpenAPI server, (iii) the DE instance.

Once the simulated network instance is up and running, the DE starts calling periodically the MEC Location API to find out the average number of users in a given zone. Apart from integrating with MEC Sandbox, the DE has various other roles such as exposing the collected metrics from the MEC Location API to the Prometheus server, acting as the custom-made OpenAPI client to request a scaling request, and defining the threshold for the triggered actions.

The experimental setup in MEC Sandbox is configured with the 4g-5g-wifi-macro scenario, four stationary users, four low-velocity users, and four high-velocity users. The DE calls the MEC Location API to calculate the average number of users of zone 3 and define the threshold to trigger actions through the custom-made Kubernetes OpenAPI. Then, if the average number of users is greater than $\Gamma$ (e.g., $\Gamma >= 3$), the DE upscales the VoD application pod accordingly (e.g., from 1 to 2 replicas). Otherwise, if the average number of users goes below $\Gamma$, the DE downscales the VoD pod replicas (e.g., from 2 to 1 replica).

The proposed framework is fundamentally different from Kubernetes native rule-based scaling (horizontal pod autoscaling). To begin with, we can use service/user-based metrics with the current framework. Also, the framework is not limited to scaling, and the applications running on the edge can directly interact with MEC APIs and RAN, providing added value to both application developers and mobile operators.

III. SUMMARY

In this work, we target microservice scaling using MEC Sandbox. The developed decision engine gathers user location information leveraging the ETSI MEC Location API for performing microservice scaling on the network edge. The implemented framework can also interact with different MEC APIs, enabling highly configurable application deployment on the mobile edge. As a future work, we propose an ML-based DE, which can allocate the edge resources proactively.

REFERENCES

[1] S. Barrachina-Muñoz, M. Payaró, and J. Mangues-Bafalluy, “Cloud-native 5G experimental platform with over-the-air transmissions and end-to-end monitoring,” in CSDSP, Porto (Portugal). IEEE, 2022.
[2] S. Barrachina-Muñoz, I. Baranda, M. Payaró, and J. Mangues-Bafalluy, “Intent-Based Orchestration for Application Relocation in a 5G Cloud-native Platform,” in NFV-SDN (in press). IEEE, 2022.
[3] R. Nikbakht, S. Kahvazadeh, and J. Mangues-Bafalluy, “Video on demand streaming using RL-based edge caching in 5G networks,” in 2022 IEEE Conference on Standards for Communications and Networking (CSCN’22), Thessaloniki, Greece, 2022.