OREOS: Demonstrating E2E Orchestration in 5G Networks with Open-Source Components

Conference Paper · May 2023

DOI: 10.1109/INFOCOMWKSHPS57453.2023.10226044

14 authors, including:

Noé Godinho
University of Coimbra

Paulo Duarte
Capgemini

David Perez Abreu
University of Coimbra

Bruno Mendes
Capgemini Engineering

OREOS: Demonstrating E2E Orchestration in 5G Networks with Open-Source Components

Noé Godinho*, Paulo Duarte†, Paulo Martins†, David Abreu‡*, Raul Barbosa†, Bruno Mendes†, João Fonseca†, Marco Silva*, Marco Araújo†, João Silva†, Karima Velasquez*, Bruno Sousa*, Marilia Curado*‡, Adriano Goes†
*University of Coimbra, Centre for Informatics and Systems of the University of Coimbra,
Department of Informatics Engineering
†Capgemini Engineering, Porto
‡Pedro Nunes Institute, Coimbra

***Abstract*—5G aims to support ubiquitous connectivity, ultra-Reliable Low Latency (uRLLC), and massive device communication in Next Generation networks. To achieve these objectives, the Open-Radio Access Networks (O-RAN) alliance aims to decouple the Radio Access Network (RAN) architecture and allow heterogeneity. To ensure the services' requirements, it is necessary to guarantee solutions that improve the management of the network. This work proposes an End-to-End (E2E) orchestration framework for a 5G communication infrastructure with open-source components. An overview of the implemented architecture is presented and two demonstrations are shown: how RAN and Core Network metrics are retrieved using a monitoring xAPP, and how the orchestrator enforces a policy after processing and analysing the data gathered. The results show that it is possible to deploy the proposed architecture to monitor and allocate resources efficiently in near-Real Time (near-RT) environments. The major novelty of this work is the fact that this constitutes the first E2E 5G network system using open-source tools, to the best of our knowledge. For this purpose, an interface adapter was built to interlink some of these open-source components.**

I. Introduction

5G and beyond aim to operate a new generation of services by transitioning to a flexible architecture that ensures heterogeneity [1]. To achieve this objective, the O-RAN alliance's purpose is to decouple and softwarise the RAN architecture [2] in an intelligent manner.

To provide efficient usage of the network resources, it is necessary to automate the orchestration of the network. In 5G environments, a MANagement and Orchestration (MANO) component must be considered [3]. MANO must decide how services are placed according to retrieved metrics, available resources, and services' requirements.

This paper proposes an E2E orchestration implementation using open-source components in the context of the Orchestration and Resource optimization for rEliable and lOw-latency Services (OREOS) project [4]. OREOS aims to enable reliability and low latency in 5G environments by designing and implementing a platform capable of ensuring the services' requirements and providing algorithms for orchestration and resource management in the network.

The authors wish to acknowledge and show their public appreciation towards the COMPETE 2020, Portugal 2020 - Operational Program for Competitiveness and Internationalization (POCI), European Union's ERDF (European Regional Development Fund) for providing the necessary co-funding of the project OREOS: Orchestration and Resource optimization for rEliable and lOw-latency Services (POCI/01/0247/FEDER/047222) and CISUC R&D Unit - UIDB/00326/2020 or project code UIDP/00326/2020. Some authors are doctoral students funded by Capgemini Engineering and, in other cases, by the Portuguese funding institution FCT - Foundation for Science and Technology. Noé Godinho wishes to acknowledge the FCT grant 'DFA/BD/7293/2020'.

II. OREOS Architecture

The OREOS project is comprised of five components, as per Figure 1: 1) RAN; 2) CORE Network (CN); 3) Mobile Edge Computing (MEC), 4) MANO, and 5) Monitoring, which envision an E2E solution in a virtualised environment.

The RAN is composed of two modules. First, as a South-Bound Interface, emulators and simulators developed by the OpenAirInterface alliance [5] are used for User Equipment (UE) and New Generation Base Station (gNB). As a North-Bound Interface, Flexible RIC (FlexRIC) [6] is used to provide the intelligence to the RAN, where the radio management algorithms are implemented, and to improve xAPPs execution. In the CN component, Open5GS [7] and ViNGC [8] are implemented, while in the MEC component, ENSCONCE [9] is used. To communicate between RAN and CN, N2 and N3 interfaces are used. The MANO component in the OREOS architecture is implemented with Open Network Automation Platform (ONAP) [10]. Since there is no native communication between FlexRIC and ONAP, a REST API has been developed to ensure connectivity and policy enforcement between FlexRIC and the orchestrator. To retrieve and analyse metrics on the network, Prometheus [11] is used jointly with Grafana [12] to provide a complete monitoring solution.

III. Demonstrations

Two demonstrations are presented in this section. The first demonstration aims to show how metrics are retrieved on RAN and CN components by executing a monitoring xAPP. The second demonstration aims to show the policy enforcement after processing and analysing the previously gathered data. The presented results are obtained in a standard scenario to provide a proof of concept for the OREOS project.

A. Metric Retrieving

Based on the O-RAN E2 Service Model (E2SM) Key Performance Matrix (KPM), FlexRIC offers Medium Access

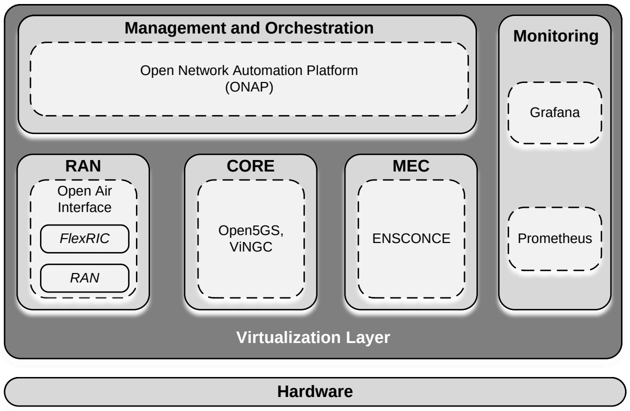

Fig. 1. OREOS Architecture Overview

Control (MAC), Radio Link Control (RLC), and Packet Data Convergence Protocol (PDCP) statistics monitoring service types. This Service Model enables the RAN Intelligent Controller (RIC) to subscribe to and obtain pertinent statistics regarding the aforementioned layers from the gNB. The obtained statistics from the FlexRIC monitoring reports are saved in a log file called *log.txt*. Figure 2 shows a snippet of the metrics retrieved from the monitoring xAPP working on a 5G environment.

```
mac_stats: tstamp=1673605658623491,frame=617,slot=12,
           dl_aggr_tbs=80707,ul_aggr_tbs=1782402,...
rlc_stats: tstamp=1673605658624693,txpdu_pkts=4,
           txpdu_bytes=12,txpdu_wt_ms=0,...
mac_stats: tstamp=1673605658626373,frame=617,slot=19,
           dl_aggr_tbs=80707,ul_aggr_tbs=1782426,...
pdcp_stats: tstamp=1673605658628164,txpdu_pkts=0,
            txpdu_bytes=0,txpdu_sn=0,rxpdu_pkts=4,...
```

Fig. 2. Logs retrieved from a monitoring xAPP

Other important metrics for the uRLLC services, such as radio latency, are also retrieved. These metrics are used afterwards by the MANO component through the connecting interface. On the MANO side, all the information will be processed, and policies will be generated according to the received data. Then, these policies will be applied by the orchestration and policy enforcement.

B. Orchestration and Policy Enforcement

When the Software Management Orchestrator (SMO) detects a situation where RAN Service Level Specifications (SLS) cannot be fulfilled, non-Real Time (non-RT) RIC can use A1 policies to improve or mitigate the situation.

A Quality of Service (QoS) optimisation policy is then generated to optimise the RAN by allocating resources or creating a specific slice. For example, the amount of resources allocated to a Network Slice is defined at Network Slice Instantiation. Still, due to high traffic demand, it is detected

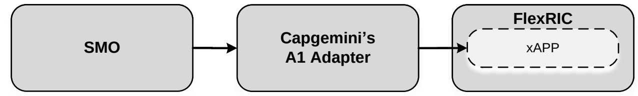

Fig. 3. Interaction between components to enforce a policy

that the resources available are not enough. So, depending on the SLS towards the slice compared with other configured services, the operator can reconfigure the allocated resources. However, that is always a risk that the service cannot fulfil the requirements defined by the SLS. In this situation, the operator can send a policy to prioritise the users by disconnecting some users of that slice to guarantee the QoS.

Figure 3 shows the components that are key to enforcing baseline policies to improve the usage of network resources. According to the gathered traffic from an xAPP, it is possible to improve the bandwidth usage and allocate the requested resources in near-RT. The type of policies and the schema related to policies must be acknowledged by the SMO so that it can create policies that an xAPP will consume on the RIC, in this case, FlexRIC.

IV. Conclusion

This paper demonstrates the retrieving of metrics and policy enforcement in an E2E orchestration architecture with open-source components for 5G. The results show how it is possible to use open-source components to monitor and allocate resources efficiently in near-RT environments within the context of 5G communications, thanks to the novel and proprietary A1 Adapter that interlinks the open-source tools.

References

[1] M. Agiwal, A. Roy, and N. Saxena, "Next Generation 5G Wireless Networks: A Comprehensive Survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 1617–1655, 2016.

[2] A. S. Abdalla, P. S. Upadhyaya, V. K. Shah, and V. Marojevic, "Toward Next Generation Open Radio Access Networks: What O-RAN Can and Cannot Do!," *IEEE Network*, vol. 36, no. 6, pp. 206–213, 2022.

[3] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, and H. Flinck, "Network Slicing and Softwarization: A Survey on Principles, Enabling Technologies, and Solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2429–2453, 2018.

[4] OREOS, "Moving forward to ultra-reliable and low latency solutions." https://oreos.pt/, accessed 11/01/2023.

[5] OpenAirInterface Alliance. https://openairinterface.org/, accessed 11/01/2023.

[6] FlexRIC, "Flexible RAN Intelligent Controller." https://gitlab.eurecom.fr/mosaic5g/flexric/, accessed 11/01/2023.

[7] Open5GS, "Open Source implementation for 5G Core and EPC." https://open5gs.org/, accessed 11/01/2023.

[8] ViNGC, "Virtualized Next Generation 5G Core." https://capgemini-engineering.com/be/en/brochure/virtualized-next-generation-5g-core/, accessed 11/01/2023.

[9] ENSCONCE. https://builders.intel.com/docs/networkbuilders/unlocking-edge-performance-with-the-capgemini-engineering-ensconce-intelr-smart-edge-open-platform.pdf, accessed 11/01/2023.

[10] ONAP, "Open Network Automation Platform." https://www.onap.org/, accessed 11/01/2023.

[11] Prometheus. https://prometheus.io/, accessed 11/01/2023.

[12] Grafana. https://grafana.com/, accessed 11/01/2023.