Orchestration of Mission-Critical Services over an NFV Architecture

Aitor Sanchoyerto, Rubén Solozabal, Bego Blanco, Elisa Jimeno, Endika Aldecoa, Estrella Basurto, Fidel Liberal

▶ **To cite this version:**

Aitor Sanchoyerto, Rubén Solozabal, Bego Blanco, Elisa Jimeno, Endika Aldecoa, et al.. Orchestration of Mission-Critical Services over an NFV Architecture. 15th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI), May 2019, Hersonissos, Greece. pp.70-77, 10.1007/978-3-030-19909-8_6 . hal-02363852

HAL Id: hal-02363852
https://inria.hal.science/hal-02363852

Submitted on 14 Nov 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Orchestration of Mission-Critical services over an NFV architecture

Aitor Sanchoyerto[1], Ruben Solozabal[1], Bego Blanco[1], Elisa Jimeno[2] ,Endika Aldecoa[1],

Estrella Basurto[1] and Fidel Liberal[1]

[1] University of the Basque Country, Bilbao (Spain)
[2] Atos Spain SA

{aitor.sanchoyerto,ruben.solozabal,bego.blanco, endika.aldecoa,
estrella.basurto,fidel.liberal}@ehu.eus[1]
elisa.jimeno@atos.net[2]

Abstract. In the race towards 5G, NFV (Network Functions Virtualization) arises as one of the enabler technologies. The intelligent orchestration of the network becomes a key element to achieve the demanded network slicing for an efficient allocation of the available shared virtualised resources.
In this paper we propose an intelligent orchestration process of mission critical services over an NFV architecture. Mission critical services have tight requirements in terms of latency and high-availability that must be met in an end-to-end basis. Our proposal includes a monitoring system that collects performance data from the VNF (Virtual Network Function) instances in order to feed the decision-making process of the orchestrator and then elastically assign resources to the network service.
The software components that compose our deployment are presented as well as the validation scenario in which the features of the test-bed are exposed.

Keywords: NFV, Management and orchestration, MC.

1 Introduction

In the recent years, the Public Safety (PS) field has shown an increasing interest towards different forms of network sharing models in contrast to building out dedicated legacy PS networks. This interest has led 3GPP to include, from Release 11 onwards, the requirements of Mission Critical (MC) communications as a central topic to address the key requirements of the next generation broadband PS networks.

In order to support the demanding set of requirements of the PS vertical, NFV offers a new way to design, deploy and manage networking services. As a key technology for the development of the 5G ecosystem, NFV decouples the network functions from proprietary hardware appliances in order to be run in a virtualized infrastructure. This architecture drives the rapid development of new network services with elastic scale and

automation. Additionally, the use of end-to-end network slicing mechanisms will allow sharing the infrastructure among other vertical industries or services and customising its capabilities on a per-tenant basis.

This paper presents the intelligent orchestration process of a Mission Critical Push-To-Talk (MCPTT) service over an NFV architecture. MCPTT compels tight requirements that include, among others, high availability and reliability, very low latency, support for one-to-one and group calls, talker identification and high audio quality for clear interchange of information. In this context, the network orchestrator must provide the tools to share edge computing capabilities between mission critical and commercial users.

MCPTT is a PS mission-critical voice communication type, aimed at the coordination of emergency teams that are organized in groups [1]. It provides an arbitrated method by which two -or more- users may engage in communication. Users may request permission to transmit (e.g., traditionally by means of a press of a button) and the MCPTT service provides a deterministic mechanism to arbitrate between requests that are in competition (i.e., Floor control). When multiple requests occur, the determination of which user's request is accepted, and which users' requests are rejected -or queued- is based upon a number of characteristics (including the respective priorities of the users in contention).

The challenge consists on transparently and elastically allocating the available resources to the variety of actors requiring different services with different priorities in space and time. To that aim, the network slicing based on virtualization techniques must be used to modify the network behaviour by changing functions or reconfiguring parameters. At this point, the intelligent orchestration process of the network slices, including commercial and MCPTT users, involves the collection of performance data through a monitoring system that processes them and generates the corresponding alerts.

This paper is organized as follows. Next section provides a brief overview on the current state of network management and orchestration towards the definition of service slices. Then, Section 3 describes the deployment of an MCPTT service over NFV architecture, including the technical specifications of the platform that will host the service. Section 4 describes the validation scenario to later, in Section 5, propose an intelligent orchestration process that includes a monitoring module to feed the decision-making engine. Finally, Section 6 summarizes the main contributions and poses new research challenges that will be addressed in the future.

2 Related Work

The previous section remarks the importance of network slicing in NFV architectures to simultaneously provide a multitude of diverse services over a common underlying physical infrastructure. The concept of slicing in 5G and next-to-5G environments has been widely discussed in the recent years.

Authors in [2] define the concept of *slicing* as the separation of a single physical network into multiple isolated logical networks. Another definition in [3] states that a network slice instance is formed by a set of network functions and logical resources enabling the deployment of a complete logical network infrastructure capable of accommodating the requirements for a specific service. Then, the network slicing in an NFV architecture includes the allocation of the common virtualised resources among the existing services.

Work in [4] shows an example of a network slicing use case over NFV with two tenants, each of them operating several service slices. The authors describe the multiple challenges to be addressed in order to make the slicing process possible. These challenges include the design of adequate resource management mechanisms that enable resource sharing among slices when necessary, without violating their required performance levels. The work in [5] proposes an orchestration system that realises application-aware end-to-end slices on demand, each of which contains not only guaranteed end-to-end bandwidth resources (i.e., in the forms of virtual links, virtual switches, and virtual access points) but also isolated Information Technology (IT) resources (i.e., in the form of virtual network functions) to carry specific applications with quality of service (QoS) guarantees. These examples reflect the interest on network slicing, but the challenge of the elastic resource allocation that dynamically distributes the available virtualized resources among the existing slices instances still remains unsolved, especially when related to the dynamic prioritization of services.

One fundamental issue when addressing the dynamic allocation of resources for network slicing in NFV is related to the possibility of scaling the VNFs. There are many algorithms that employ machine learning or other Artificial Intelligence (AI) methods to decide the optimal number of instances of a VNF needed to provide the requested service. The study in [6] targets the dynamic provisioning of network services expressed as one or multiple service chains in cloud datacentres and designs efficient online algorithms without requiring any information about future traffic rates. Authors in [7] seek a proactive approach to provision new instances for overloaded VNFs ahead of time based on the estimated flow rates. The authors formulate the VNF provisioning problem in order that the cost incurred by inaccurate prediction and VNF deployment is minimized. The work in [8] addresses a dynamic VNF service chain deployment and scaling by a novel combination of an online provisioning algorithm and a multi-armed bandwidth optimization framework, which exploits online learning of the available bandwidths to enable the optimal deployment of a scaled service chain. This kind or methods solve the analytical problem of deciding when and how to scale a VNF, but must jet be introduced into a complete orchestration system in order to be integrated with the management and orchestration (MANO) operation.

The existing literature shows that, although network slicing has gained a great interest, the proposed solutions are still in process of being fully developed. Among the many challenges that arise, one fundamental issue is the automation of the resource allocation for the elastic dimensioning of the slice. The following sections of this paper introduce the concept of service monitoring, that involves the collection and processing of performance data from the NFV architecture. The monitoring system is an intelligent decision-making module that generates the necessary alerts to trigger the orchestration

processes, such as VNF scaling, in order to perform the re-allocation of resources as needed.

3 Deployment of Mission Critical Push-To-Talk over an NFV architecture

To evolve the MCPTT into an NFV-ready service it must be defined as a Network Service and, as such, be described by a Network Service Descriptor (NSD). Each VNF that composes the service chain, for its side, is defined by a VNF Descriptor (VNFD) plus the image that implements its correspondent functionality. When a MC service is instantiated, the orchestrator must create the network slice that accommodates critical services ensuring the high availability, reliability, and low latency expected for these communications. This implies not only the allocation of the necessary resources to instantiate and run the VNF chain, but also the monitoring and scaling of the service in order to meet the elastic demand. The MC slice must have a prioritized access to resources in the cloud infrastructure, which may result in a low-priority service deallocation, if required, in order to meet the service level agreements (SLAs).

The specifications of the platform and the virtualized service are described in the following sections.

3.1 Platform specifications

The platform selected to perform the testbed complies the ETSI NFV standard [9]. The orchestrator selected is Open Source MANO (OSM) [10], an ETSI-hosted initiative for the development of open-source NFV Management and Orchestration software stacks that is aligned with the ETSI reference architecture. OSM works in conjunction with Canonical's Juju to configure the VNFs. The Virtual Infrastructure Manager (VIM) used in the project is OpenStack.

3.2 Service specifications

Presenting the service as a single VNF provides little flexibility for scaling possibilities. The service must be separated into functional blocks that belong to different VNFs composing the whole network service. This allows an efficient orchestration of the service according to the demand and at the same time it enables the provision of redundancy to the service.

Key to define the VNFs that compose the service is to detect which ones belong to the signalling plane and which ones to the data plane. Bear in mind that data plane is susceptible to scale up proportionally to the number of users while the control play may follow a different scaling ratio. Being able to separate these planes, enables us to replicate the data plane near the end-user easily, following the demand in specific locations.

Figure 1 shows how MCPTT service has been divided in several VNFs following the user/control plane separation criteria, as well as considering the most efficient way

to scale the service. The first elements being isolated are control elements as the DNS (Domain Name System) server, the HSS (Home Subscriber Server) and the controlling CSC (Common Services Core) servers. The CSC is composed by the services: IDMS (Identification Management System), KMS (Key Management System), CMS (Configuration Management System) and GMS (Group Management System). This enables the redundancy of these critical control components. The MCPTT application server is also separated into a functional block that manages the Floor Control responsible for the arbitration of talk turns, and the Media Distribution VNF, in charge of processing the media encryption, recording and media multiplexing tasks. This way the system achieves a complete Control and User Plane Separation. The service descriptors are defined in OSM following the Yang information model.

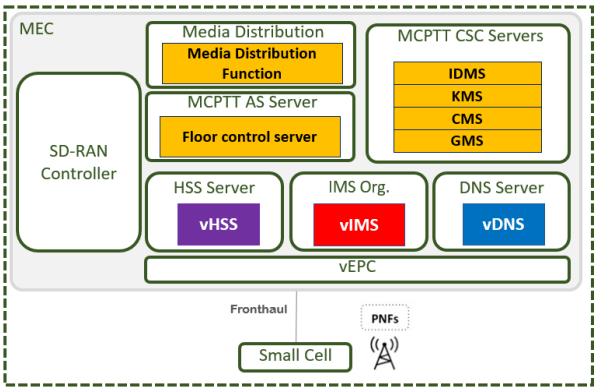

Figure 1: MCPTT network service

4 Validation Scenario

To demonstrate that this architecture provides a solution for deploying efficient and elastic MC services over a network slice, we will describe a validation scenario [11]. It is composed by three stages with an increasing emergency level, that are described next. The objective of this validation scenario is how the MCPPT service behaves with the increase in each scenario of the number of PS users that will use the MCPTT service when increasing the alarm level. Monitoring and orchestration will ensure continuity and quality of service.

Stage 1: Under normal circumstances, the platform provides different network slices. Some slices correspond to PS organizations running respectively a MC service and while other slices correspond to legacy end users that have subscribed to the classical communications. Each network slice is composed of an allocation of cloud resources. In addition to the QoS guarantees for each tenant, the deployment owner has to assure the required levels of isolation in the provisioning of the network slices.

Stage 2: In case of emergency, the MANO will be able to react to the new service requirements. For instance, the MCPTT communications provider may require additional service in order to cope with an increased number of first responders. Based on service scaling policies, the MANO will implement new elastic resource allocation schemes, giving priority access to first responders and taking into cloud resources (for deploying more resource-consuming cloud services). In this situation, the commercial slices may suffer a service degradation in favor of the PS slices in order to guarantee the quality of service of mission-critical applications.

Stage 3: In case that ICT infrastructure is damaged during a natural disaster or a terrorist attack, the infrastructure should be able to be operative to dynamically instantiate required services to ensure local service continuity until the infrastructure is restored.

5 Monitoring and Orchestration

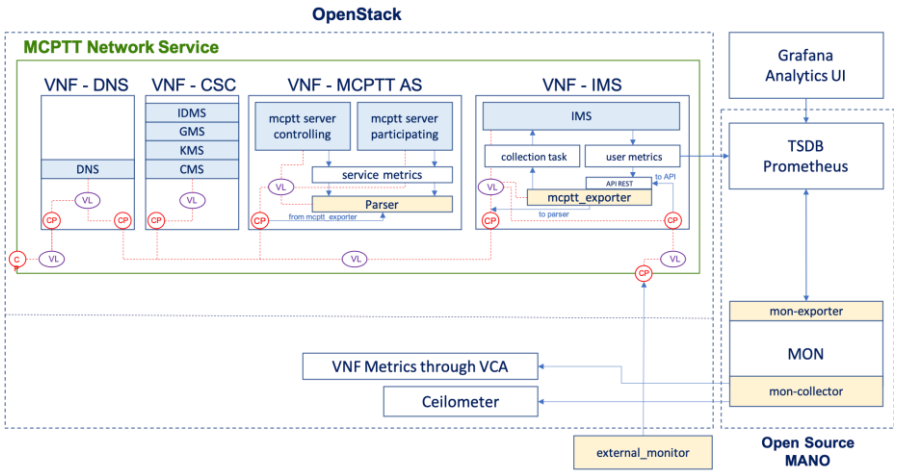

Figure 2: Service Monitoring and Orchestration

Telemetry is the automated process of collecting measurements from remote or inaccessible equipment and transmit them to receivers. Monitoring is a continuous assessment that is in charge of the analysis of the data extracted from telemetry and use it to manage the service. The telemetry and monitoring in an NFV system are mainly in charge of managing information that the hypervisor or the VNFs itself expose during execution. These systems are expected to enable additional non-functional requirements such as: scalability, non-intrusiveness and service continuity.

In order to guarantee the quality of the service, the monitoring system must be continuously analyzing the metrics received. These metrics can be oriented towards re-

sources or service performance indicators. With the first ones the system extracts information of the hardware infrastructure on which the services are deployed. And with the second, it ensures that it complies with the agreed SLAs.

Each service requires a specific capacity to perform efficiently, minimizing processing times and delays. The quality of the services offered to customers must comply with the service level contracted, for this reason, the monitoring system must be provided with intelligence. So that depending on the metrics received by telemetry, it can predict the degradation of the quality of service and notify the system orchestrator to perform the most appropriate decision.

Figure 2 represents how the resource metrics are collected using the Open-Stack Monitoring Frameworks [12]. Ceilometer [13] is focused on collecting data from physical and virtual resources available to the cloud deployment. These data can be stored by a Gnochi [14] database, and actions or alarms can be triggered according to predefined conditions set by the user on Aodh [15].

On the other hand, service metrics are exported using Juju´s charms, preconfigured with the VCA (VNF Configuration and Abstraction) on the orchestration phase. With this procedure, the VNF can be accessed to gather any information from the running services, in this case scenario these metrics are being collected by the mcptt_exporter service.

Then, the service metrics are collected using Prometheus [16], which is a white box monitoring and alerting system designed for large and scalable environments that includes built-in and active scraping, storing, querying, graphing, and alerting based on time series data. Prometheus covers the domains of instrumentation (using so called "exporters" to instrument platform or applications) and monitoring (providing mechanisms for the data collection, alerting, etc.). It features:

- A multi-dimensional data model, where data can be sliced and diced along multiple dimensions like host, service, endpoint and method.
- Operational simplicity to set up monitoring anywhere without being an expert through configuration files.
- Scalable and decentralized, for independent and reliable monitoring.
- A powerful query language that uses the data model for meaningful alerting and visualisation.

As mentioned, an exporter must be introduced into the monitored service. It is in charge of publishing all the information related to the users, calls and from the MCPTT Server itself. These service metrics are collected and stored by Prometheus, while Grafana is used for graphic visualization.

The data collected during monitoring can be used to trigger events for automated resource re-provisioning, in cooperation with the orchestrator OSM. One of the main methods to guarantee the fulfillment of the SLAs is autoscaling. This solution needs the allocation of a load balancer to monitor and distribute the loads across all the VMs on the scaling group. Hence, autoscaling can increase, decrease, and replace instances without manual intervention even across thousands of instances. This should not be confused with scale-in and scale-out of the Network Service; NS scaling refers to the

addition or removal of VNFs. OSM supports scale-out and scale-in operations on running services.

6 Conclusions and Future Work

This paper presents an NFV architecture that deals with the tight requirements of a Mission-Critical service. It is capable of maintaining the high availability and reliability required for the service, while sharing the infrastructure in an elastic manner between mission critical and commercial users thanks to the supervision of a network orchestrator. To that aim, the network slicing based on virtualization techniques is key to maintain isolation between clients.

The intelligent orchestration process of the network slices involves the collection of performance data through a monitoring system that processes them and generates the corresponding action on the infrastructure. Being the most significant the scaling of the service VNFs. There are many algorithms that employ machine learning or other AI methods to decide the optimal number of instances of a VNF needed to provide the requested service. This paper is focused on presenting the architecture, foreseen for the future the definition of the metrics or the development of an intelligent decision-making system that cooperates with the orchestrator.

Acknowledgements. This work has been partly funded by the EU funded H2020 5G-PPP project ESSENCE (Grant Agreement Nº 761592) and the Spanish Government's MINECO project 5GRANVIR (TEC2016-80090-C2-2-R).

References

[1] *Technical Specification Group Services and SystemAspects; Mission Critical Push to Talk (MCPTT) over LTE*, 3GPP TS 22.179 v16.1.0 Stage 1(Release 14), April 2018.

[2] Foukas, X., Elmokashfi, A., Patounas G., Marina, M.K. , "Network Slicing in 5G: Survey and Challenges" *IEEE Communications Magazine*, vol. 55, no. 5, pp. 94-100 (2017). DOI: 10.1109/MCOM.2017.1600951.

[3] Martínez, R., Vilalta, R., Casellas, R., Muñoz, R., Fei, L.,Tang, P., López, V., "Network Slicing Resource Allocation and Monitoring over Multiple Clouds and Networks. In proceedings Optical Fiber Communications Conference and Exposition (OFC)", pp. 1-3 *IEEE* (2018).

[4] Ordoñez-Lucena, J., Ameigeiras, P., Lopez, D., Ramos-Munoz, J.J., Lorca, J., Folgueira, J., "Network Slicing for 5G with SDN/NFV: Concepts, Architectures and Challenges", *IEEE Communications Magazine*, vol. 55, no. 5, pp. 80-87 (2017). DOI 10.1109/MCOM.2017.1600935.

[5] Han, K., Li, S., Tang, S., Huang, H., Zhao, S., Fu, G.,Zhu, Z.,"Application-Driven End-to-End Slicing: When Wireless Network Virtualization Orchestrates With NFV-Based Mobile Edge Computing", *IEEE Access*, vol. 6, pp. 26567-26577, 2018. DOI: 10.1109/ACCESS.2018.2834623.

[6] Wang, X., Wu, C., Le, F., Liu, A., Li, Z., Lau, F., "Online VNF Scaling in Datacenters." *IEEE 9th International Conference on Cloud Computing (CLOUD)*, San Francisco, CA, pp. 140-147. (2016). DOI: 10.1109/CLOUD.2016.0028.

[7] Fei, X., Liu, F., Xu, H., Jin, H., "Adaptive VNF Scaling and Flow Routing with Proactive Demand Prediction", *IEEE Conference on Computer Communications*, Honolulu, HI, 2018, pp. 486-494. DOI: 10.1109/INFOCOM.2018.8486320.

[8] Wang, X., Wu, C., Le, F., Lau, F. C. M., "Online Learning-Assisted VNF Service Chain Scaling with Network Uncertainties", *IEEE 10th International Conference on Cloud Computing (CLOUD)*, Honolulu, CA, pp. 205-213 (2017). DOI: 10.1109/CLOUD.2017.34.

[9] ETSI Industry Specification Group, "Network Functions Virtualization (NFV); Virtual Network Functions Architecture", Etsi GS NFV-SWA 001, 2014.

[10] Open Source Mano: "*https://osm.etsi.org/".*

[11] 5G ESSENCE Deliverable D2.2, Overall System Architecture and Specifications, June 2018.

[12] OpenStack Telemetry: *"https://wiki.openstack.org/wiki/Telemetry".*

[13] OpenStack Celiometer: *"https://docs.openstack.org/ceilometer/".*

[14] OpenStack Gnocchi: *"https://wiki.openstack.org/wiki/Gnocchi".*

[15] OpenStack Aodh: *"https://docs.openstack.org/aodh/*".

[16] Prometheus: *"https://prometheus.io/".*