NFV 환경에서 적응적 스트리밍을 위한 관리 및 오케스트레이션 시스템에 관한 연구

Linh Van Ma(*), 최근창(**), 김진술(***)
(* ** ***) 전자컴퓨터공학과, 전남대학교
(*) linh.mavan@gmail.com, (**) t_o_p1412@naver.com, (***) jsworld@chonnam.ac.kr

A Study on Management and Orchestration System for Adaptive Streaming in NFV

Linh Van Ma(*) GeunChang Choi(**), Jinsul Kim(***)
(** ***) School of Electronics and Computer Engineering, Chonnam National University*

Abstract

In this research, we present a study on the management system to control adaptive streaming servers. We use the management system to start and stop the servers based on current system load to optimize resources. We propose several metrics to monitor resources. We implement the system using python, the adaptive streaming servers using node.js. Besides, we run all of the system components in Docker to virtualize a cloud streaming system. The result indicated the system reduce resource consumption up to 10%.

1.Introduction

Network function virtualization (NFV) [1] describes how to virtualize and manage network resources, such as storage, and computing in a network. The virtual network functions manager (VNFM) [2] is a fundamental component of the NFV management and organization (MANO) architectural framework. The VNFM works in concert with other NFV-MANO [2] functional blocks to increase the interoperability of software-defined networking elements and help standardize the functions of virtual networking. VNFs is an important essential to carry out the business benefits outlined by NFV architectural. They provide real network functions as real network components do but it requires VNFM to manage virtualized components. VNFMs are very necessary for the expansion, operational changes, add new resources, and inform other VNFs function blocks in the architecture NFV-MANO. In this paper, we exploit the MANO structure to build a video streaming system. The MANO manages streaming servers. It can start/stop a server depending on the current load of the system. In this way, we can manage network resource efficiently.

2.SystemOverview

In the streaming system, we have three main components. First, a DASH management server is responsible for listening requests from clients and directing those requests to content servers based on selecting algorithms. Secondly, an uploading and transcoding server makes uploaded files from clients to DASH content. Finally, DASH content delivery network

(CDN) servers respond requests coming from clients and provide video streaming to clients.

The main server collects measurement information of CDN DASH servers to redirect a request from a client to the best serving server. We use seven parameters to evaluate the performance of a DASH streaming server. Regarding local measurement, we have; 1) Working CPU, 2) Free memory, 3) Total memory, 4) Load average. Network measurement, we have; 5) Ping, 6) Throughput upload, 7) Throughput download. DASH servers use this information for the purpose of retrieving DASH files from other servers in case it does not serve a transcoding request from a client. This technique is a solution for sharing DASH files of a server when it contains a multimedia video while other servers do not have the video content. A server can retrieve DASH files from other servers based on metric information from the available parameters. For example, Server A and Server B has the same DASH video, and Server A is faster than B 25%. If Server C does have the video content, it can get 75% video DASH files from A and 25% DASH files from B.

We arrange three physical upload servers with mediocre performance, two physical CDN DASH servers to provide video streaming. Each server can start several Node.js applications. Consequently, we calculated resource consumption by summarizing the total CPU performance and RAM usage in a period of all physical computers. We set up two scenarios to compare the system performance between the system having Docker, NFV orchestrator and the system having applications which only run the physical servers. In client, we upload several files with different resolutions up to 4K. We then measure the system performance around 30 seconds in the both scenarios. As a result, the system managed resource efficiently up to 10% compared to systems without management.

3.Conclusion

In this paper, we discussed the current issue emerged in today virtual network technologies. Our study contributed two main points. First, we discussed about network virtualization technologies. Secondary, we implemented DASH streaming system using MANO architecture.

Acknowledgments

This research was supported by the IT R&D program of MSIT(Ministry of Science and ICT), Korea / NIPA(National IT Industry Promotion Agency) 12221-14-1001, Next Generation Network Computing Platform Testbed. Also, this work (Grants No. C0513295) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2017.

References

[1] Mijumbi, Rashid, Joan Serrat, Juan-luis Gorricho, Steven Latre, Marinos Charalambides, and Diego Lopez. "*Management and orchestration challenges in network functions virtualization.*" IEEE Communications Magazine Vol 54, No. 1, pp. 98-105, 2016.

[2] Giotis, Kostas, Yiannos Kryftis, and Vasilis Maglaris. "*Policy-based orchestration of NFV services in Software-Defined Networks.*" In Network Softwarization (NetSoft), 2015 1st IEEE Conference on, pp. 1-5. IEEE, 2015.