Student thesis proposal: Design and implementation of auto-scaling solution for 5G user plane functions

November 30th, 2019

1 Objective

Investigate and develop novel scaling strategies that can be applied to/improve the auto-scaling of user plane functions (UPFs) in the Tieto's 5G testbed.

2 Background

Tieto's 5G testbed is a testbed for 5G research in Tieto. It represents a real-world telecommunication system in a virtualized/cloud environment and allows developing and optimizing solutions for end-to-end telecommunication infrastructure services in emerging 5G mobile networks. More specifically, the testbed is deployed using a cloud management platform named OpenStack [1]. The testbed components run as virtual machines (VMs) on top of an OpenStack infrastructure. Scaling (or auto-scaling) is among the most crucial aspects of any system. The scaling can be classified into horizonal scaling, which adds/removes nodes to the resource pool and vertical scaling, which increases/decreases the capacity of the existing node. The horizontal scaling is one of the stringent requirements for the Tieto's 5G testbed.

Existing scaling solutions for a cloud-based system are usually based on resource utilization of VMs such as CPU load, disk, memory, etc. For example, OpenStack Heat [2] - an OpenStack Orchrestration project - has already a support for a threshold-based autoscaling of VMs using CPU load reported from OpenStack monitoring tools such as Ceilometer or Monasca. However, the resource utilization is not an accurate estimation of the actual load of high performance packet process applications such as 5G UPFs. Therefore, a new method which uses the actual traffic load information is required to have a better scaling decision.

3 Research Work

- Investigate VNFs/SFCs, standards applied for them, and a simple deployment of VNFs/SFCs
- Fundamentals of workload analysis and load-based scaling solutions
- Revisit the existing load-based scaling solutions (incl. the RECAP solution, or the ones in the literature)
- Implement a simple load-based scaling solution and integrate with the VNF Manager to trigger the creation/deletion of VMs
- Extend the proposed solution with one of the following options:
  (1) Evolve the solution to be smarter with Fuzzy logic and reinforcement learning [3, 4, 5], or other machine learning techniques.
  (2) Combine load-based auto scaling + load balancing
- Performance evaluation using Tieto data / data generator (with a simple traffic model or traffic replay)

4 Requirements

- Prerequisites: C/C++/Python programming, basic OpenStack, Linux
- Deliverables: a plugin-based, tightly coupled with the Tieto testbed
- Duration: 5-6 months

References

[1] Openstack. https://www.openstack.org/.

[2] Openstack. https://wiki.openstack.org/wiki/Heat.

[3] Hamid Arabnejad et al. An auto-scaling cloud controller using fuzzy q-learning-implementation in openstack. In *European Conference on Service-Oriented and Cloud Computing*, pages 152–167. Springer, 2016.

[4] Hamid Arabnejad et al. A comparison of reinforcement learning techniques for fuzzy cloud auto-scaling. In *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pages 64–73. IEEE Press, 2017.

[5] Abdelquoddouss Laghrissi et al. A fuzzy logic-based mechanism for an efficient cloud resource planning. In *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–7. IEEE, 2019.