Building Blocks for an Elastic Mobile Core

Lirim Osmani‡, Heikki Lindholm‡, Binoy Chemmagate‡, Ashwin Rao‡,
Sasu Tarkoma‡, Johanna Heinonen†, Hannu Flinck†
‡University of Helsinki, †Nokia Networks

ABSTRACT

The mobile core is currently served by gateways having a convoluted control plane and data plane. We abstract and modularize the key functions of these gateways, and separate the control and data plane in nwEPC, an open source implementation of the mobile access network gateways. We use our custom load balancer for the control plane to demonstrate the elasticity offered by our implementation.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: [Network topology, Wireless communication, Network communications]

Keywords

Mobile Core; Network Functions Virtualization; Software Defined Networking

1. INTRODUCTION

The mobile core is lagging behind data centers in abstracting and modularizing the key functions of its gateways. Such modularization is essential for deploying programmable and manageable network functions, and consequently make the mobile core elastic.

In this paper, we demonstrate our solution of separating the control and data plane in nwEPC, an open source implementation of the mobile access network gateways, namely, the Serving Gateway (SGW) and the Packet Data Network Gateway (PGW). We then leverage on the separated control and data plane and build a rudimentary load balancer for the control plane. Preliminary results show that our load balancer can be used to manage the load of at least four SGWs.

The need to simplify the convoluted mobile core arises from not only the ever increasing volumes of mobile data traffic, but also the signaling traffic which is expected to grow 50% faster than the data traffic [4]. In their position paper Li *et al.* [11], highlight the brittleness of the cellular infrastructure, and present the benefits of an SDN managed mobile core. Basta *et al.* [5], abstract the key functions of these gateways, and also discuss the benefits of separating the control and data plane in these gateways. Jin *et al.* [8], build on these insights, and sift the mobile Internet traffic at the base stations and group the flows based on the ISP policies. Such grouping enables aggregation of forwarding rules thus simplifying the forwarding tables in the intermediate switches. In their seminal work aimed at separating the control and data plane, Kempf *et al.* [9], propose extensions to OpenFlow [2, 12] for supporting the GPRS Tunnelling Protocol (GTP).

In the rest of the paper, we present an overview of the key network functions in the mobile core. We then present a brief description of our implementation and conclude by discussing our vision for our work.

CoNEXT Student Workshop'14, December 2, 2014, Sydney, Australia.
Copyright 2014 ACM 978-1-4503-3282-8/14/12 ...$15.00.
http://dx.doi.org/10.1145/2680821.2680832.

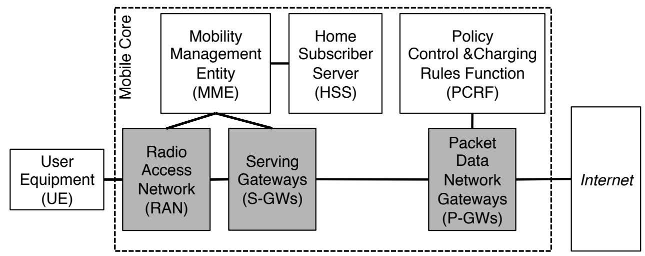

Figure 1: Key network functions in the mobile core. *The blocks in gray serve the control and data plane of the traffic.*

2. THE INELASTIC MOBILE CORE

In Figure 1, we present the key network functions in the mobile core, specifically, the Evolved Packet Core [1]. We believe that similar functions will be used in the future and that the legacy interfaces will be preserved for backward compatibility with existing equipment.

The mobile device, or User Equipment (UE), uses the Radio Access Network (RAN) to connect to the Internet through the mobile core. The Serving Gateways (SGWs) are the local mobility anchor, responsible for the handovers between the RAN access points when the UE moves, while the Packet Data Network Gateways (PGWs) are the entry point to the Internet. Currently, only one SGW serves a UE, and a UE can be served by multiple PGWs. The Mobile Management Entity (MME) is responsible for session and subscriber management and the Home Subscriber Server (HSS)

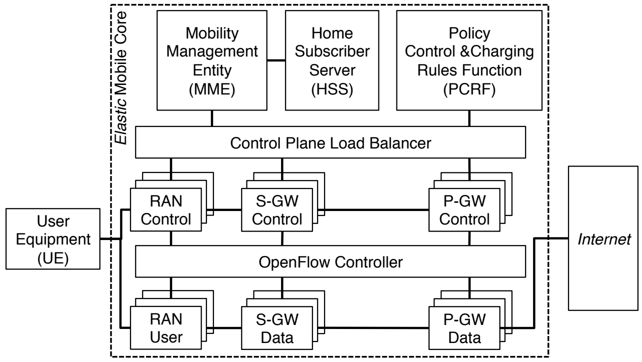

Figure 2: Separating the control plane and data plane of SGW and PGW. *Multiple instances of the control and data plane instances of the SGWs and PGWs can be launched to serve the varying network load.*

contains the subscription details of each user. The Policy and Charging Rules Function handles policy decisions and charging rules such as differentiated levels of Quality of Service (QoS) for each UE.

To summarize, the RAN, the SGWs, and the PGWs, serve the control plane and data plane of the traffic. The control and data plane for the RAN has been split by Gudipati *et al.* [6]. We now present our solution for separating the control and data plane in the SGWs and the PGWs.

3. MODULARIZING THE SGW AND PGW

We implemented a prototype of a modular gateway for the mobile core by separating the control and data planes of the open-source nwEPC gateway implementation. The control plane issues commands to the data plane using JSON-RPC over TCP, and not unlike OpenFlow [2], our command set supports addition, deletion and modification of flows. We are currently extending this JSON-RPC interface, for example, to support commands related to service differentiation and traffic metering with triggers.

In Figure 2, we present the architecture of an elastic mobile core. The separation of control plane from data plane opens the possibility to get, in real-time, a detailed picture of the network state, and this can be leveraged upon to efficiently manage the control plane signaling and data plane forwarding.

As a first step to demonstrate the benefits of modularizing the SGW and PGW, we implement a simple load balancer for the control plane of the SGW. There are numerous signaling messages between the MME and SGW [1, 4], and when there are multiple SGW control instances, there should be a load balancer to redirect these control signals to stateful SGW instances. Furthermore, this load balancer must adhere to the strict time constraints associated with these signals [1].

To evaluate our implementation, we measured the performance of single gateway (GW) instances, both non-split, monolithic version and the control plane side of the split GW, and for comparison, the load balancer (LB) with a single GW behind it and four GWs behind it with varying traffic loads. The test cases comprised of registering new user sessions at predefined rates; 75k sessions were registered per GW.

Reg./Second	500	1000	2000	4000	8000
GW-m CPU%	4.56	8.88	17.45	35.28	63.25
GW-c CPU%	21.27	40.47	81.27	X	X
LB (1) CPU%	1.93	3.87	7.70	X	X
LB (2) CPU%	1.88	3.48	6.64	13.69	26.65

Table 1: CPU usage for different load levels of the monolithic GW (GW-m), control plane of the split GW (GW-c), load balancer with 1 GW (1) and load balancer with 4 GWs (2).

Table 1 summarizes our results. A single monolithic nwEPC GW can realistically handle up to 8000 registrations per second (rps) with CPU load averaging 63% in our setup, whereas the split gateway, due to increased signaling, currently handles 2000 rps with CPU load averaging 81%. The split GW is still unoptimized, but we also note that data plane load would reduce the monolithic GW's control plane performance whereas the split GW would be unaffected. The load balancer could easily scale up to four GWs (and probably beyond) and 8000 rps with average CPU load of 27%. Given the round-robin load balancing algorithm used, the number of gateways behind the load balancer did not affect its CPU usage in practice. Latency appeared to be a possible problem as the signaling protocol times out in two seconds and timeouts were observed even with relatively low CPU load of less than 30%. We are currently evaluating the timeliness and scalability of our system.

4. DISCUSSION AND FUTURE WORK

Heinonen *et al.* [7], demonstrate another possibility that splitting the control and data planes opens, namely, free choice of data plane transport path. They show that data plane traffic can be switched dynamically to go through either hardware or software switches. Similarly, the Open Evolved Packet Core (OpenEPC) [3] is also aimed at splitting the control plane and data plane of the gateways in the mobile core.

We plan to extend our system with the ability to dynamically react to varying work loads by growing and shrinking the number of GWs. Reacting to the load level requires real-time feedback from the GWs and logically centralized decision making. Splitting the load balancer into control plane and data plane parts seems like the logical step forward, but it is not clear if splitting is feasible without compromising efficiency. For example, Levin *et al.* [10] show that a centralized (both logically and physically) controller can become the bottleneck of the system. We believe that the data plane part of load balancing could be moved to SDN controlled switches, similar to what Wang *et al.* [13] demonstrate, and we plan to evaluate the current OpenFlow solutions in this respect.

5. REFERENCES

[1] 3GPP - The Mobile Broadband Standard. http://www.3gpp.org/.

[2] Open Flow. https://www.opennetworking.org/sdn-resources/onf-specifications/openflow.

[3] OpenEPC - Open Evolved Packet Core. www.openepc.net.

[4] Signaling is growing 50% faster than data traffic. http://nsn.com/index.php?q=system/files/document/signaling_whitepaper_online_version_final.pdf, 2012.

[5] Basta, A., Kellerer, W., Hoffmann, M., Hoffmann, K., and Schmidt, E.-D. A Virtual SDN-Enabled LTE EPC Architecture: A Case Study for S-/P-Gateways Functions. In *Proc. of IEEE SDN for Future Networks and Services* (2013).

[6] Gudipati, A., Perry, D., Li, L., and Katti, S. SoftRAN: Software defined radio access network. *Proc. of HotSDN* (2013).

[7] Heinonen, J., Partti, T., Kallio, M., Lappalainen, K., Flinck, H., and Hillo, J. Dynamic Tunnel Switching for SDN-based Cellular Core Networks. In *Proc. of the 4th ACM Workshop on All Things Cellular* (2014).

[8] Jin, X., Li, L. E., Vanbever, L., and Rexford, J. SoftCell: Scalable and Flexible Cellular Core Network Architecture. In *Proc. of ACM CoNEXT* (2013).

[9] Kempf, J., Johansson, B., Pettersson, S., Luning, H., and Nilsson, T. Moving the mobile Evolved Packet Core to the cloud. In *Proc. of IEEE WiMob* (2012).

[10] Levin, D., Wundsam, A., Heller, B., Handigol, N., and Feldmann, A. Logically Centralized?: State Distribution Trade-offs in Software Defined Networks. In *Proc. of HotSDN* (2012).

[11] Li, L. E., Mao, Z. M., and Rexford, J. Toward Software-Defined Cellular Networks. *2012 European Workshop on Software Defined Networking* (2012).

[12] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., and Turner, J. OpenFlow: Enabling Innovation in Campus Networks. *ACM SIGCOMM Computer Communication Review* (2008).

[13] Wang, R., Butnariu, D., and Rexford, J. Openflow-based server load balancing gone wild. In *Proceedings of the 11th USENIX Conference on Hot Topics in Management of Internet, Cloud, and Enterprise Networks and Services* (Berkeley, CA, USA, 2011), Hot-ICE'11, USENIX Association, pp. 12–12.