Siren: A Platform for Deployment of VNFs in Distributed Infrastructures

Lyndon Fawcett
Lancaster University
School of Computing and Communications
l.fawcett1@lancaster.ac.uk

Nicholas Race
Lancaster University
School of Computing and Communications
n.race@lancaster.ac.uk

ABSTRACT

Fog computing is conceiving an Internet where general purpose compute is ubiquitous, in turn this is providing new infrastructures for Network Functions Virtualisation (NFV). However, current NFV designs focus on the Cloud, resulting in broken and suboptimal deployments when deploying to the Fog. Through a case study with preliminary results, this paper presents the effectiveness of *Siren*: a new prototype platform designed as a tool to deploy and manage Virtual Network Functions in Fog environments.

1 INTRODUCTION

In the ever-changing landscape of the Internet, the concept of Fog Computing [1, 2, 8] has recently emerged. This proposes the distribution of general compute power throughout the Internet, bringing the Cloud closer to the end user in compute locations known as *Micro-Clouds* [3] or *Nano-Datacenters*. These can provide lower latency, reduced traffic to the core, and increased scale and can consist of a range of devices from low-powered Customer Premises Equipment (CPE) at the edge of the network, through to higher-powered servers within telephone exchanges.

The benefits of this environment could be brought to the concept of Network Functions Virtualisation (NFV) by deploying VNFs to the Fog [8] through NFV Infrastructures (NFVIs). In realising a *Fog-NFVI*, we gain a number of advantages; i) Fog devices are significantly closer to the end point, avoiding potential WAN latency issues, ii) Fog devices can supplement the overall NFV compute pool, addressing some of the concerns around the general scalability of NFV in the real-world [6], and iii) as the Fog supports execution within a variety of locations, issues such as the privacy of sensitive data and information can be mitigated.

However, current deployment architectures such as Management and Orchestration (MANO) are designed to run VNFs within resource-rich, homogeneous, centralised, and well maintained cloud data centres [6]. This is in contrast to the Fog, where compute is heterogeneous and resides within a variety of locations with limited maintenance and resources. Using a Cloud-focused MANO to orchestrate VNFs in the Fog can lead to extreme configuration challenges, suboptimal placement of VNFs, and potentially network or service downtime. In part, this is because these systems are typically not concerned with the location that they are deploying to.

There are a number of challenges that need to be addressed before these concepts can be brought together, including the unsuitability of current management platforms, the consideration of the unique heterogeneity of resources, and the optimal placement of services in distributed environments. The work described in this paper focuses towards solving the aforementioned challenges addressing aspects of multi-tenancy, orchestration, monitoring, and tactical placement. The remainder of the paper describes the system architecture of Siren as well as a virtualised Content Delivery Network (vCDN) case study with preliminary results.

2 SIREN OVERVIEW

In order to address the challenges previously described, we present *Siren* a platform designed to handle the inherent variability seen in Fog environments. The architecture, shown in Figure 1, uses multiple components to realise network services within a highly dynamic environment. All code and further evaluation details are available on GitHub[1].

Infrastructure Discovery. This module is responsible for discovering devices and maintaining state and contextual information regarding those connected. Siren's Fog devices use the Docker engine [7] in order to execute VNFs. Upon initialisation, its container *Agent* automatically reports resources, network context, and capabilities to the *Infrastructure Discovery* module which is accessible by the *provisioner*.

Provisioner. The *Provisioner's* role is to provide knowledge about the infrastructure, to realise orchestrator requests,

SOSR'17, April 3–4, 2017, Santa Clara, CA, USA

DOI: http://dx.doi.org/10.1145/3050220.3060611

[1] https://lyndon160.github.io/Siren-Provisioner/

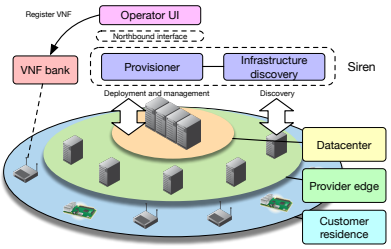

Figure 1: Siren architecture

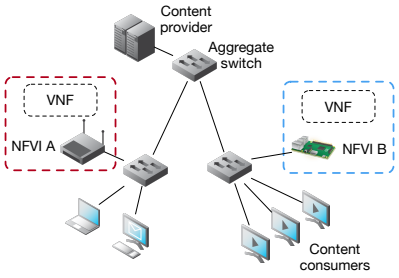

Figure 2: Evaluation topology

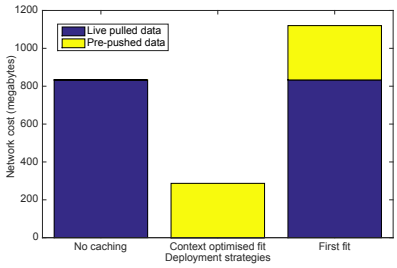

Figure 3: Caching results

and to manage the lifecycle of services. The northbound RESTful interface of the provisioner provides an abstraction of the infrastructure to orchestration applications such as the *Operator Interface* or our previously created auctioneer [4]. These applications can then use this information to make decisions about what and where services should be placed. Currently this supports the remote Docker API [7]. Docker swarm is used to manage scalability management issues.

Operator Interface. This is one of many applications that interface with the NBI of the provisioner to observe the network and virtualisation infrastructures, and assist an operator to make decisions on where the service should be placed. Service requests are then passed to the *Provisioner* in the form of a TOSCA specification describing the service. Importantly, users of this platform, including the content provider in the later case study, can request for network based tests between clients and Fog nodes.

3 CASE STUDY: VIRTUAL CDN

The benefits of combining Fog computing and NFV can be shown through a vCDN; vCaches can be deployed closer to the edge, reducing network traffic whilst improving client's quality of service [5].

Figure 2 shows the simple experiment topology that is representing two home networks that share an aggregate switch to the Internet. This case study compares the cost to the network in three scenarios: 1) a vCache VNF being deployed using a first fit clustering policy 2) a vCache VNF being deployed using an context-optimised policy and 3) using no CDN and requesting video directly from content provider source. The vCache is deployed to the NFVI (Raspberry Pi) as a Docker container. The video in use for the case study is a 1080p version of the Big Buck Bunny standard testing video and is 276.1MB in size. During each experiment iteration, three clients are watching the video and traffic is being monitored and recorded on the aggregate switch to evaluate the difference between deployment techniques. In the in second test, where optimised placement is used, the content provider requests information about latency between the service customers and the available NFVIs to determine the closest one to deploy to.

3.1 Preliminary results

Figure 3 shows the results of three VNF placement techniques. The stacked bar chart shows live data which is data that was pulled over the aggregate switch whilst clients were watching the video, and pre-pushed data which is the cost of pushing the vCache VNF. In this instance we can see that the contextual information provided by siren allowed for the VNF to be placed much closer to the client on NFVI B, thus reducing observed traffic on the aggregate switch by two thirds when compared to no caching. More importantly, a suboptimal placement policy such as first fit which in this instance places the VNF on NFVI A can cost more to the network when compared with no caching, this is due to the VNF being pushed but and the content to being requested over the aggregate switch.

4 CONCLUSION AND FUTURE WORK

This work highlights the importance, challenges, and applicability of NFV for the Fog. Furthermore, it presents a prototype system and its operational behaviour for managing and orchestrating network services in a Fog environment. Siren is evaluated using a case study and preliminary results that show significant reduction in network cost. Future work will include evaluating with different orchestration techniques and further comparisons against existing systems.

REFERENCES

[1] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli. Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*. ACM, 2012.

[2] O. Consortium and A. Working. OpenFog Reference Architecture for Fog Computing. (February), 2017.

[3] Y. Elkhatib and et al. On using micro-clouds to deliver the fog. 2016.

[4] L. Fawcett, M. H. Broadbent, and N. J. P. Race. Combinatorial auction-based resource allocation in the fog. 2016.

[5] N. Herbaut, D. Negru, Y. Chen, P. A. Frangoudis, and A. Ksentini. Content delivery networks as a virtual network function: a win-win isp-cdn collaboration. In *Global Communications Conference, 2016 IEEE*, 2016.

[6] R. Mijumbi, J. Serrat, J. L. Gorricho, and N. Bouten. Network Function Virtualization: State-of-the-art and Research Challenges. (c), 2015.

[7] Solomon Hykes. Docker. docs.docker.com.

[8] L. M. Vaquero and L. Rodero-Merino. Finding your Way in the Fog. *ACM SIGCOMM Computer Communication Review*, 2014.