

NI: A Benchmark Suite for NfvInsight

ABSTRACT

Keywords

Network Function Virtualization;

1. INTRODUCTION

Network function virtualization (NFV) has become a hot topic, both in industry and academia. Since the publication of NFV Introductory White Paper [1] of ETSI in 2012, a lot of works have been emerged in this field. Modification works of NFV were done on the whole software stack (or maybe both SW and HW stack?). There are de facto industrial NFV platform OPNFV [2] as well as advanced NF allocation frameworks like OpenNF [3], CoMb [4] and E2 [5]. Also, there are works like OpenBox [6] and Netbricks [7] to rewrite or modify the NFs.

However, a suite of easy-to-use NFV benchmark is not yet existed. It is unavoidable to experience a time consuming process finding both open source software and proper chaining policies. According to our observation, most NFs used in papers are different open source implementations linked in different kinds of NF chain policies. So that, there is not yet a general baseline for measurement and comparison. Furthermore, the workload generator and network traffic trace used are also different, and real-world traces need to be provided to test different NF chains.

We also did a survey among top conference researchers who have experiences setting up NFV environment. In our survey, we found that the deploying time varies much due to the scale. In average, it takes around 1 month to build up a NF cluster having less than 10 VM instances. But when the scale of instances increase to over 50, the build up process can take 3-4 months or more. One of our respondent said that they were still keeping on iterating and improving their testbed constantly.

There are more complains pointing out their pain points: 1) Automate the setting up and testing process. 2) Configure and stabilize NFs (?). 3) Write rules to set up topology and enforce flow control.

In this paper, we develop a suite of NF benchmarks, which is supposed to have the following characteristics: 1) Representing typical NFs 2) Easy to deploy 3) Plenty metrics for measurement (please specify)

For the demand of representative NFs, we referred to the NFV Introductory White Paper [1], which defined eight scenario of NFV use cases. Then we collected open source implementations accordingly. Table 1 lists the scenarios and the information of NFs used in our benchmark. However, not only single NFs should be typical, but also the NF chains. We consulted industrial partners for the real world scenarios including (..) traditional middle boxes (and VoIP services). The typical NF chains are listed in Table 2.

For the demand of easy to deploy, we hope to achieve the goal of one-click installation, and reduce the setting up time to less than 1 week no matter the scale of the testing environment. (whats the user interface like?) We leverage Docker and Kubernetes to pack NFs in containers, manage the images, and do allocation automatically. We use OVS to do switching, and leverage pre-written scripts and Openflow rules to implement chaining and flow control.

We leverage the monitoring function of Kubernetes and provide metrics for measurement.

2. BACKGROUND AND MOTIVATION

2.1 Network Function Virtualization

2.2 Survey

[2, 1]

3. BENCHMARK DESCRIPTION

4. IMPLEMENTATION

5. EVALUATION

6. CONCLUSIONS

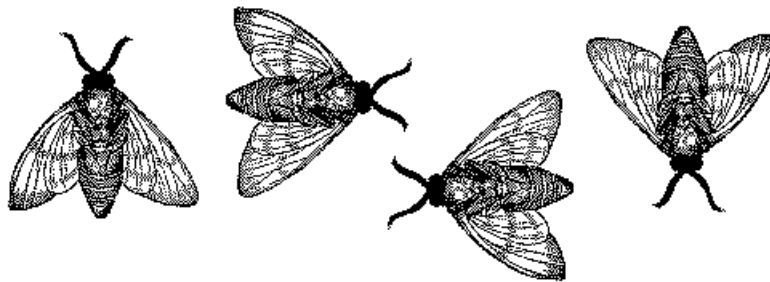


Figure 1: A sample black and white graphic that needs to span two columns of text.

Table 1: Frequency of Special Characters

Non-English or Math	Frequency	Comments
\emptyset	1 in 1,000	For Swedish names
π	1 in 5	Common in math
\$	4 in 5	Used in business
Ψ_1^2	1 in 40,000	Unexplained usage

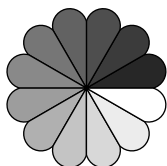


Figure 2: A sample black and white graphic that has been resized with the `includegraphics` command.

7. ACKNOWLEDGMENTS

8. REFERENCES

- [1] L. Lamport. *LaTeX User's Guide and Document Reference Manual*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1986.
- [2] S. Salas and E. Hille. *Calculus: One and Several Variable*. John Wiley and Sons, New York, 1978.

Table 2: Some Typical Commands

Command	A Number	Comments
<code>\alignauthor</code>	100	Author alignment
<code>\numberofauthors</code>	200	Author enumeration
<code>\table</code>	300	For tables
<code>\table*</code>	400	For wider tables