DevOps for Software-Defined Telecom Infrastructures

draft-unify-nfvrg-devops-01

C. Meirosu, A. Manzalini, J. Kim, R. Steinert, S. Sharma, G. Marchetto, I. Pappafili

UNIFY is co-funded by the European Commission
DG CONNECT in FP7

Introduction

- Significant work on network virtualization technologies and their management in both academia and industry fora (several WGs in IETF and IRTF, ETSI NFV ISG, TMForum)
- Purpose of the document: discussion opener in NFVRG
  - describe a set of principles that are relevant for applying DevOps ideas to managing software-defined telecom network infrastructures
  - identify challenges related to developing tools, interfaces and protocols that would support these principles and leverage standard APIs for simplifying operations tasks

DevOps thinking for software-defined infrastructures

- Roles
  - Service developer
  - VNF developer
- DevOps principles
  - Deploy with repeatable, reliable processes
    - Identical, trusted processes; visibility
  - Develop and test against production-like systems
    - Isolation properties; frequent customizations
  - Monitor and validate operational quality
    - Established procedures for validation – for example Y.1564
  - Amplify feedback loops
    - Mainly corporate-cultural aspects

Original principles from: "DevOps, the IBM Approach" 2013

Challenge areas

- Stability of the software-defined infrastructure versus continuous changes
- Consistency, Availability and Partitioning trade-offs
- Observability: scalability, distribution, automation
- Verification: when to do, what to check, scalability
- Troubleshooting: automated workflows
- Identification and definition of performance metrics

Stability

- An adaptive routing system (hopefully) is stable most of the times
  - individual protocols designed to reduce instability intervals
- Some typical causes of instability
  - failures
  - configuration errors
  - protocol design: dependency between state definition and update mechanisms
- In software-defined infrastructures, orchestrators and controllers create a multitude of inter-related control loops
- Control loops need to work correctly
  - in the presence of failures
  - variable delays in communicating with a distributed set of devices
  - frequent changes in network conditions

Consistency, Availability and Partitioning

- CAP theorem general idea: two of three properties can be achieved simultaneously
- Which combination is most appropriate for software defined infrastructure in operators, in which scenario?
  - CA
  - AP
  - CP
- Example of relationship change with SDN
  - Standard routing: Availability implies Consistency. control plane Partition usually implies data plane Partition.
  - SDN routing: not necessarily (nodes continue to forward already installed flows for a while when losing control plane connection)
- CAP choices challenging to coordinate for basic virtual resources and VNFs (seen as resources)

Observability

- Trade-off between network observability and scalability (overhead, intensity, # mon. instances, etc.)
- Distributed operation and information exchange between monitoring functions
- Configurability and conditional observability, in particular with self-adaptive algorithms
- Automation
  - mapping of monitoring functionality from a logical forwarding graph to virtual or physical instances executing in the infrastructure
  - (re-)placement of monitoring

Verification

- Machine-readable descriptions of configurations help (YANG, etc)
- Can we expect orchestration and controllers to export or log every change?
  - Integration within the orchestrators and controllers
- Verification of middle-boxes a huge challenge
  - State explosion

Troubleshooting

- Automated workflows
  - interfaces needed permeating all layers
  - automatic building
  - possibility for on-demand manual intervention
- Use of active measurements
  - Complex configuration and interpretation of results
  - Scalability (both in terms of generated traffic, and use of resources for network, computation and storage)

DevOps Performance Metrics

- Technical considerations directly related to the service provided
  - already defined by IETF and other relevant bodies
- Process-related considerations regarding the deployment, maintenance and troubleshooting of the VNFs
  - number of probes for end-to-end QoS monitoring
  - delay between service order and deliver
- Cost-related considerations associated to the benefits from using a Software-Defined Telecom Infrastructure
- Measure-ability, availability of data sources

Next steps

- Comments and feedback are welcome
- Further refine the DevOps principles and their application in the NFVRG context
- Improve the description and positioning of the challenges
- Identify areas where use cases should complement challenges description
- Aim for follow-up discussion at IETF 93

The research leading to these results has received funding from the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement no. 619609 - the UNIFY project. The views expressed here are those of the authors only. The European Commission is not liable for any use that may be made of the information in this document