Container-based NFV:
Opportunities and Challenges

Richard Cziva - University of Glasgow, United Kingdom
Richard.Cziva@glasgow.ac.uk
GEANT TESTBED SERVICE WORKSHOP @NORDUnet 21/10/2015
About me

• PhD candidate at University of Glasgow since 2013
  • Member of Networked Systems Research Laboratory (Netlab) http://netlab.dcs.gla.ac.uk

• Main research interests:
  • Cloud resource management
  • Network Function Virtualization
  • Software-Defined Networking

• Going to be an intern at NORDUnet for 4-5 months
  • Measurement / performance verification for GTS
Middleboxes

- Hardware-based network appliances that manipulate network traffic
  - Firewall
  - Load balancer
  - VPN
  - Intrusion Detection and Prevention Systems
  - WAN Accelerator
  - Web cache

- Enterprise networks rely on middleboxes
  - Middleboxes represent 45% of the network devices
  - The advent of customer devices will further increase the number
Problems with middleboxes

- They incur significant capital investment
- They are cumbersome to maintain
- They can not be extended to run new functionality
- The run proprietary software
  - limits innovation
  - creates vendor lock-in
Network Function Virtualization

- NFV decouples network functions from the hosting platform
- Can reduce capital and operational expenditure
- Improves resource efficiency
- Introduces fault-tolerance and scalability
- Works well with Software-Defined Networking
State of the art

- OpenStack: early stage demos for NFV
- OPNFV: Linux foundation project, first release “Arno” is out
- Cloud4NFV: VM-based NFV orchestration for private clouds
- ClickOS: a custom, high-performance XEN-based VM
- “Stateless network functions”
- …
Issues

• Operator specific implementations

• Poor reuse of software components
  • Deploy and configure once for a specific server
  • Operator specific deployment system(s)

• Inability to create/destroy network functions quickly
  • Inserting routing rules and deploying + configuring software is complex
  • Costly operation

• VMs used as NFs introduce a high overhead

• Lack of scalability
Glasgow Network Functions

• Glasgow Network Functions (GLANF)
  • research and development project
  • Est. 2014

• Main characteristics:
  • Container-based
  • Transparent
  • Infrastructure independent
  • Open innovation

• Two key contributions of GLANF are
  • Using containers for NFs
  • End-to-end transparent traffic management (using SDN)
Containers

- Lightweight “virtualization”
- Fast create/start/stop/delete
- High performance
  - Small delay, high throughput, low memory footprint
- Reusable / Shareable
- Traditional software environment
- Microservice architecture
GLANF design

• Router
  • Hosted on the Open Daylight Controller
  • Creates and inserts the rules to apply a specific forwarding policy

• Manager
  • Provides a REST API to the system

• Agent
  • Daemon running on the GLANF servers
  • Manages containers and local routing
  • Provide host/container status information

• UI
  • Talks to the Manager
  • Adds/removes network functions
Step-by-step

• Traffic from Server1 to Server4

• Need a new Firewall placed between them?
  • Controller find a GLANF server
  • Pull the firewall image
  • Spawn an instance

• Apply the policy
  • Reroute the traffic matching:
    • FROM server1
    • TO server2

• Chaining Containers
  • Web Cache
  • IDPS
Inside the GLANF Server
Demo

- https://youtu.be/W7aa4L2piBQ (4:49)
Container NFV - challenges

1. Exclusive allocation of CPU resources
2. Direct I/O (e.g. SR-IOV)
3. Inter-NF communication (direct memory mapped)
4. High performance software switch is required on the host (maybe in a VM?)
5. Fast live migration
Challenges - performance

• If we don’t need to copy the packet from kernel space
  • Good throughput and latency
  • Examples: iptables, tc

• If we need to copy a packet to user space
Challenges - performance

• Using SR-IOV NIC for VNF Containers
  • High performance
  • H/W offloads
  • Low latency using user-mode driver

• Intel DPDK runs in Docker container (Intel 06/2015)

• Using DPDK / SR-IOV we can reach close to physical appliance
  • The trade-off is flexibility
  • DPDK and SR-IOV requires support of the NIC
Container NFs in GTS?

• Could a (container) NF be a resource in GTS in the future?

• Example use-cases:
  • Firewalls
  • Transparent measurement modules for network researchers
  • Introduce delay
  • Rate limiter
  • Load balancer
  • ?
Thank you!

- Contact: Richard.Cziva@glasgow.ac.uk

- GLANF has been published in two papers so far: