NETLAB
NETWORKED SYSTEMS RESEARCH LABORATORY

# GNFC: Towards Network Function Cloudification

**Richard Cziva**, Simon Jouet, Dimitrios Pezaros
University of Glasgow, United Kingdom
IEEE NFV-SDN, San Francisco, California, US. 2015.11.16

# Motivation

- Enterprise networks rely on middleboxes
  - 45% of network devices
  - The need for network processing will increase even further
- Public cloud computing
  - Growing adoption (36% - Forrester)
  - Migrating compute services is well supported
  - Migrating network services (network functions) is hard

# Goal

- To enable NFV for tenants of (public) clouds
- Make cloud-native NFs that support
  - Elasticity
  - Automated management
  - Scalability
  - Pay as you go
- How do we get there?
  - Advanced network programmability
  - Stateless NFs (state and functionality should be separated) that run on top of commodity VMs and hypervisors

# Related work

- CloudNaaS
  - Allows deployment of NFs in a private cloud deployment
- APLOMB
  - Outsourcing middleboxes to the cloud
- Startos
  - Network-aware orchestration for cloud middleboxes
  - Abstraction for cloud middleboxes

# Contributions of the paper

1. we looked at the existing limitations of public IaaS clouds that challenge the use of an NFV framework by tenants
2. we present GNFC, an NFV framework deployable in public clouds
3. we evaluate GNFC over Amazon EC2, Google Compute Engine and Microsoft Azure

# Limits for public cloud users

- Programmability
  - Extremely limited view and programmability
- IP forwarding
- Protocol support
  - Many protocols are filtered out or handled in a special way

| | EC2 | Azure | GCE |
|---|---|---|---|
| GRE | ✓ | ✗ | ✗ |
| VXLAN | ✓ | ✓ | ✓ |

- Support for NFV frameworks
  - Custom hypervisors or hardware can not be deployed

# Glasgow Network Functions

- GNFC: Glasgow Network Function for the Cloud
- An NFV framework
- Can be deployed over public cloud infrastructures
- Can attach various NFs to selected user traffic

- Open-source: http://glanf.dcs.gla.ac.uk

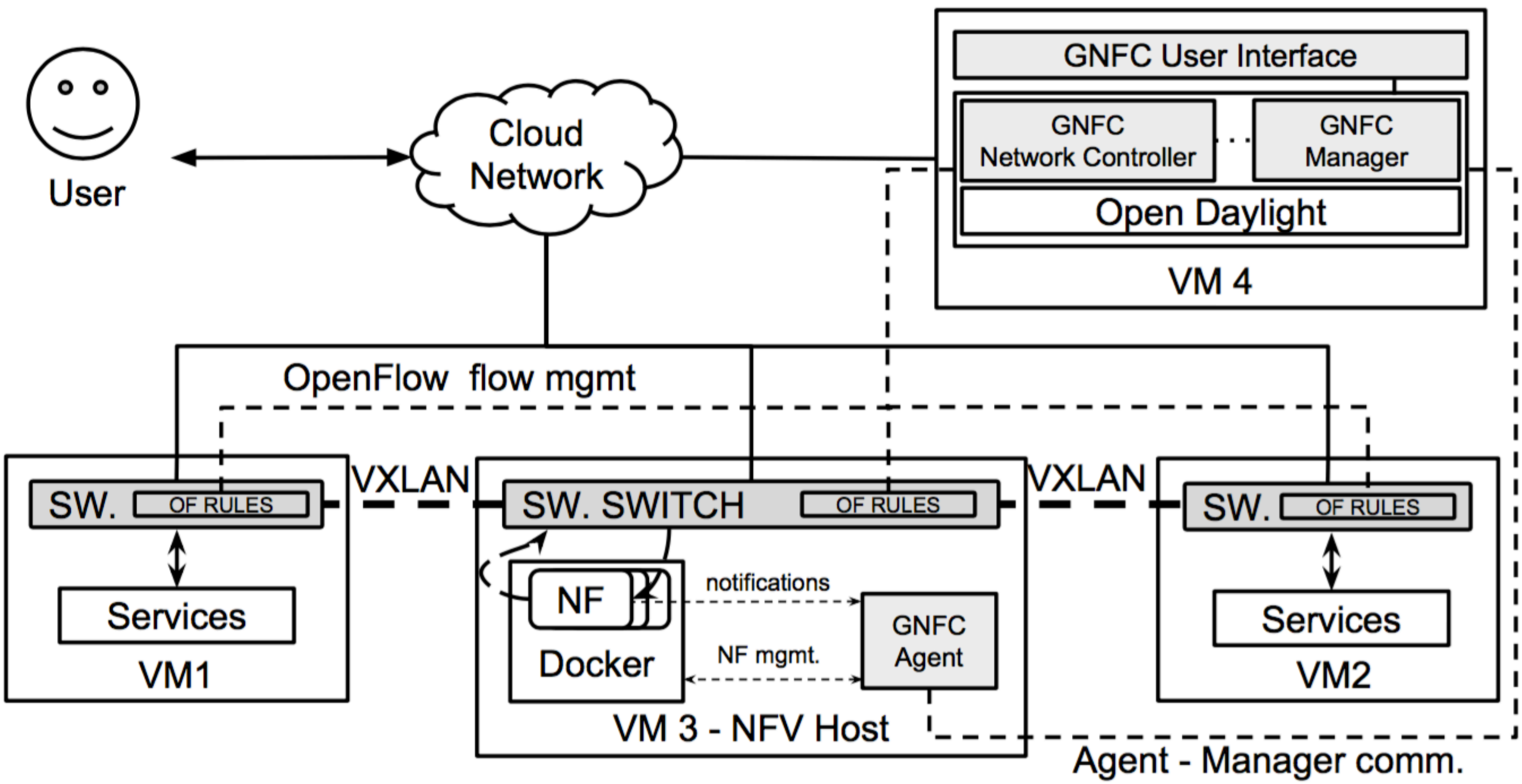
# GNFC highlights

- VXLAN overlay (tunneled) network
- OpenFlow-based SDN
- Transparent traffic redirection
  - no change in packets
- Containers instead of VMs for NFs

# Containers

- Lightweight “virtualization”
- Fast create/start/stop/delete
- High performance
  - Small delay, high throughput, low memory footprint
- Reusable / Shareable
- Traditional software environment
- Micro-service architecture
- Challenges (by Ramki Krishnan, Dell – 45 minutes ago)
  - “Can containers offer the isolation and security that VMs offer?”
  - “Several vNFs can not be hosted in containers in this moment”

# GNFC components

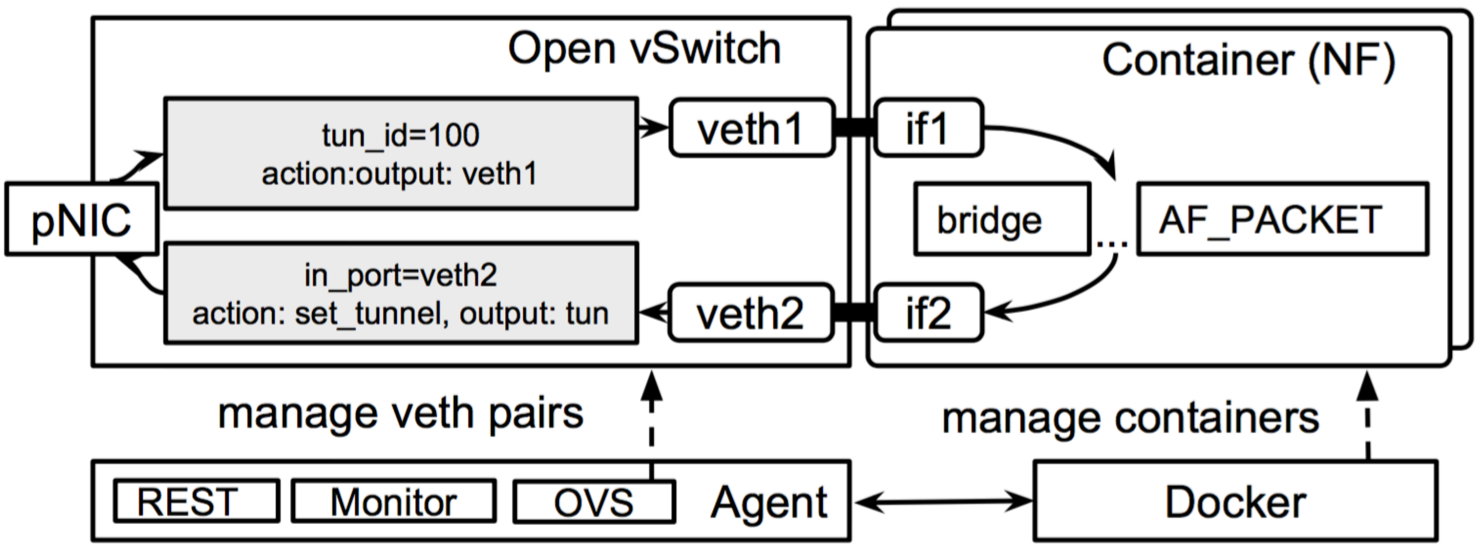


# Transparent redirection

| Hop | VM | Match | Action |
|---|---|---|---|
| 1 | VM1 | in_port=LOCAL<br>nw_dst=VM2 | set_tunnel:100<br>set_field:VM3->tun_dst<br>output:tunnel |
| 2 | VM3 | in_port=eth0<br>tun_id=100 | output:veth1 (NF1's input) |
| Middlebox processing of VM1 $\rightarrow$ VM2 packets in NF1 | | | |
| 3 | VM3 | in_port=veth2<br>(NF1's output) | set_tunnel:101<br>set_field:VM2->tun_dst<br>output:tunnel |
| 4 | VM2 | in_port=LOCAL<br>nw_dst=VM1 | set_tunnel:102<br>set_field:VM3->tun_dst<br>output:tunnel |
| 5 | VM3 | in_port=eth0<br>tun_id=102 | output:veth1 (NF1's input) |
| Middlebox processing of VM2 $\rightarrow$ VM1 packets in NF1 | | | |
| 6 | VM3 | in_port=veth2<br>(NF1's output) | set_tunnel:103<br>set_field:VM1->tun_dst<br>output:tunnel |
| 7 | VM1 | in_port=eth0<br>tun_id=103 | output:LOCAL |

# GNFC Host

- The VM that runs the NFs

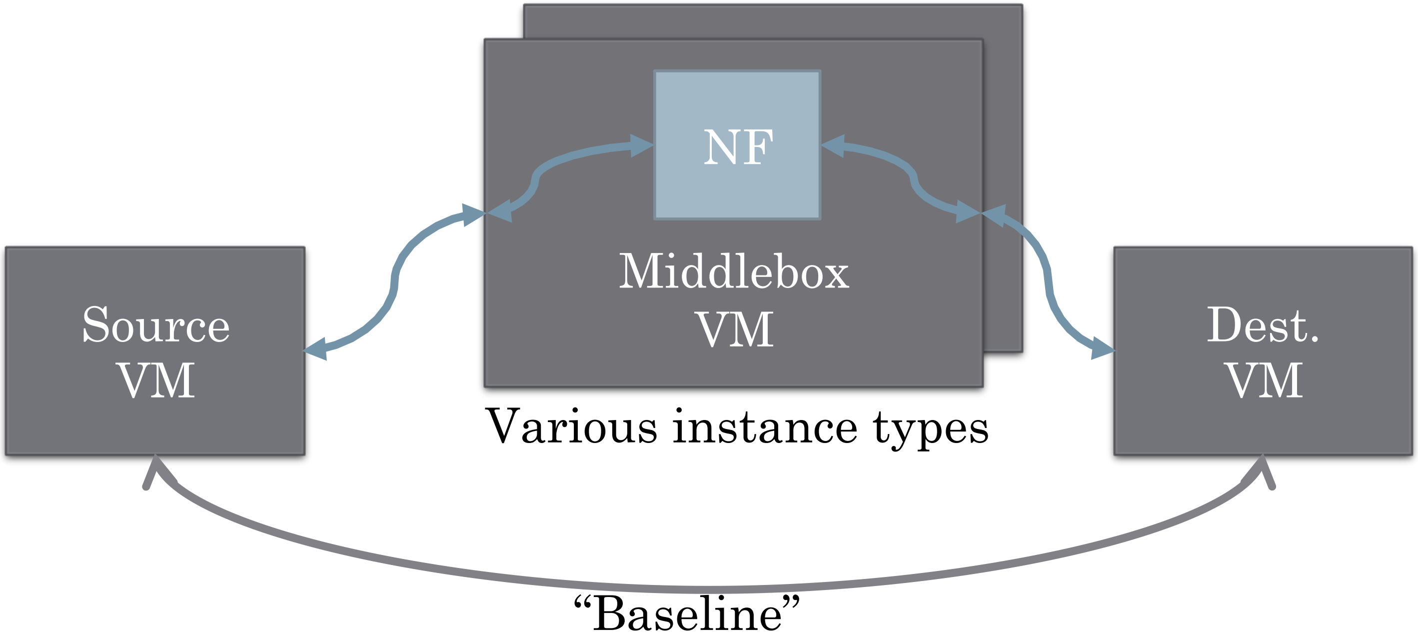


# GNFC NFs

- There are no restrictions for NFs
  - They need to forward packets between two interfaces
- Available example NFs:
  - Firewall (iptables-based)
  - Traffic control
  - HTTP filter
  - Rate limiter
  - SNORT
  - …
- Open source, all defined with Dockerfiles: https://github.com/glanf/dockerfiles

# Experimental results

- Focused on the overhead of traffic tunneling through a NF
  - We requested new VMs for middleboxes in every measurement round (10 measurements / instance type)
- The middlebox VM was running one wire NF

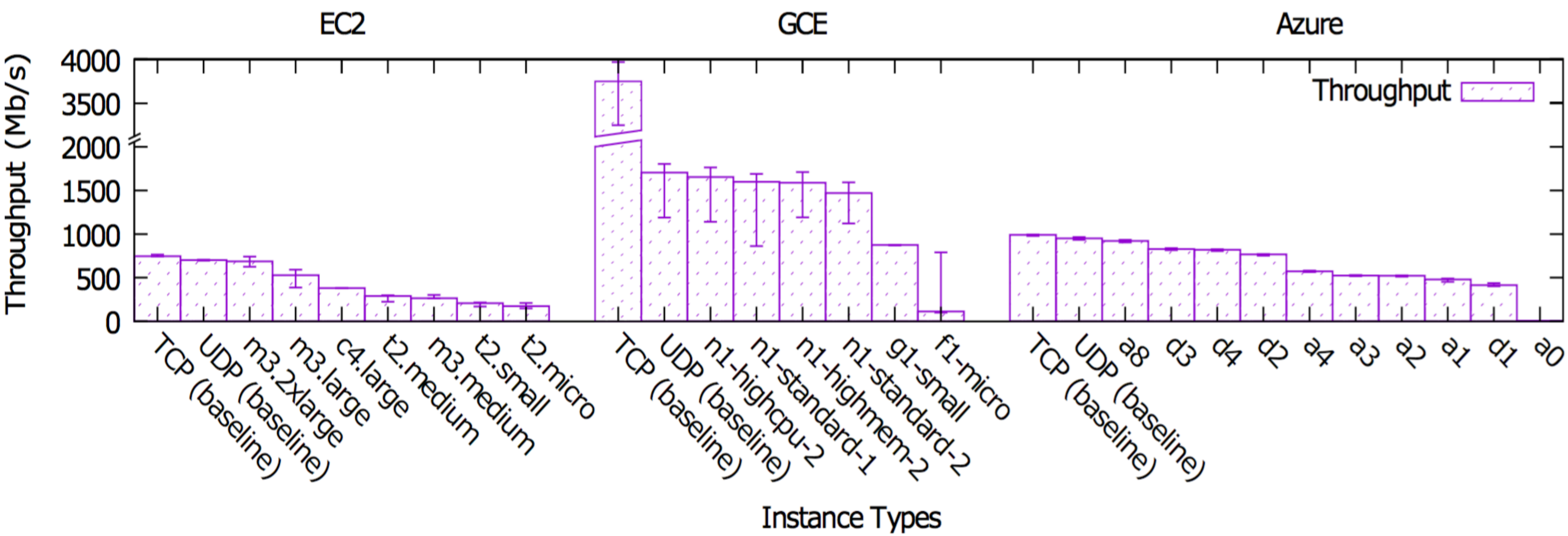


# GRE vs VXLAN (EC2 only)

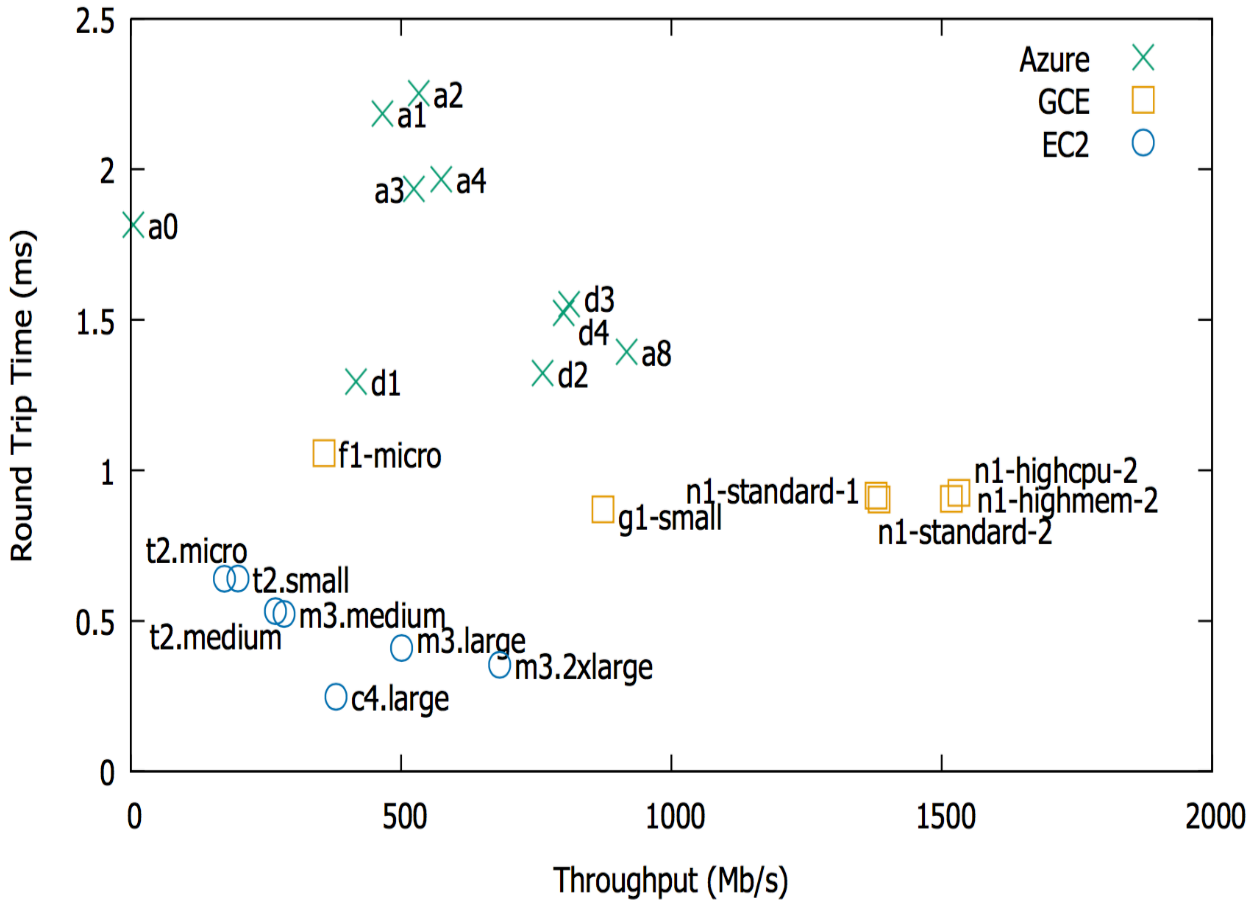
- Only Amazon EC2 supports both
- GRE performs a bit better for our measurements
  - Can be explained with a lower encapsulation overhead

| | RTT (ms) | | | Throughput (Mbit/s) | | |
|---|---|---|---|---|---|---|
| **Instance** | **GRE** | **VXLAN** | **G/V** | **GRE** | **VXLAN** | **G/V** |
| t2.micro | 0.61 | 0.65 | 0.94 | 214.50 | 215.50 | 1.00 |
| t2.small | 0.64 | 0.62 | 1.03 | 232.50 | 236.50 | 0.98 |
| t2.medium | 0.49 | 0.51 | 0.97 | 292.50 | 300.00 | 0.97 |
| m3.medium | 0.48 | 0.47 | 1.01 | 290.50 | 286.50 | 1.01 |
| m3.large | 0.39 | 0.42 | 0.93 | 608.00 | 579.00 | 1.05 |
| m3.2xlarge | 0.33 | 0.37 | 0.89 | 745.00 | 696.50 | 1.07 |
| c4.large | 0.23 | 0.25 | 0.94 | 380.00 | 380.00 | 1.00 |
| | | Average: | **0.95** | | Average: | **1.01** |

# End-to-end throughput



# RTT / throughput



# Summary of results

- Instance price / achievable throughput through the VM is not linear
- There are significant differences between the 3 major providers
- According to our measurements from May 2015

| Provider | Throughput | Latency | Variance |
|---|---|---|---|
| Amazon EC2 | low | low | low |
| Google CE | high | medium | high |
| Microsoft Azure | medium | high | low |

# Conclusion

- To enhance cloud adoption of enterprises, NFV should be supported in public clouds
- Glasgow Network Functions for the Cloud (GNFC)
  - Container-based NFV framework that can be deployed over existing public clouds
  - In public clouds we use a overlay network based on VXLAN
  - Enables basic NFV for tenants
  - Evaluated over three public clouds

# Thank you!

NETLAB
NETWORKED SYSTEMS RESEARCH LABORATORY


- Contact: Richard.Cziva@glasgow.ac.uk
- http://netlab.dcs.gla.ac.uk