Towards a
Network Marketplace in a Cloud

Da Yu, Luo Mai, Somaya Arianfar,
Rodrigo Fonseca, Orran Krieger, David Oran

HotCloud 2016
Denver, CO
“Public” clouds today

• Few providers
• Vertically integrated
• Opaque control of lower layers
Example: MapReduce on Amazon

MapReduce!

Ec2

Better MapReduce!

ElasticMr
“Public” clouds today

- Few providers
- Vertically integrated
- Opaque control of lower layers
- Vendor lock-in

Competition

Innovation

Google Cloud Platform
Amazon Web Services
Microsoft Azure
Massachusetts Open Cloud

- Open Cloud Exchange: public and open cloud
Massachusetts Open Cloud

- Open Cloud Exchange: public and open cloud

See http://info.massopencloud.org/ for more details
Where does Networking fit?

- Our goal: networking should be a first class service, not just basic infrastructure
- Providers should compete, *all the way to the physical level*, to provide networking!
Can we make networking part of the marketplace?

- **Vision:**
  - Alice wants to provision 50 VMs in a compute pool, and 50 VMs in a GPU pool
  - Wants to connect these VMs in a single LAN
  - Gets offered two options to create this network:
    - Standard, best effort network (free)
    - Dedicated 10Gbps network with RDMA (for $$)
  - Selects the second option, and her traffic is automatically routed through the right switches, with the right queues configured
Why?
Potential reasons

• **Isolation**
  – Physical (security)
  – Performance (BW/Latency)

• **Control**
  – E.g., DCTCP, PFabric, NetFPGA

• **Special/Niche needs**
  – 100Gbps, Lossless Ethernet (RoCE, RDMA), 60GHz wireless,…

Enable innovation and market-driven competition for network services
The NetEx Prototype
Baseline Architecture

Common Network

Pod A
GPUs

Pod B
Storage

Pod C
Compute

Jointly administered machines w/ internal network
Multi-Provider Inter-Pod Network
Multi-Provider Inter-Pod Network

Pod A

Pod B

Pod C

Network Exchange Marketplace

Network Users

Network Providers

NetEx Agent
Configuring a Path

- User requests *path properties* between two or more pods, for specific traffic (*flowspec*)
  - E.g., “I need 10usec latency for my port 445 RPC traffic”
- NetEx forwards request to network providers
- Network providers return bids:
  `<path segment, properties, price>`
- User selects path option
- NetEx brokers provisioning...
Configuring a Path

- Selected network provider provisions internal path segment, returns opaque handle
- NetEx configures pod switches:
  <flowspec, provider, handle>
- Requirement: packet matching flowspec arrives at the egress EoP provider port with the label handle
Current Pod-level Design

• **Current prototype in Mininet**
  – Uses OpenFlow with nested MPLS labels

• **Divides responsibility between ToR and EoP**
  – EoP directs traffic to providers based on per-provider label
  – ToR matches on flowspec and configures loose source route:
    source pod label > provider label > provider internal label > dest
fwd(A\|int)\npop(X) pop(Y)\nfwd(X\|Y) pop(A\|int)\npush(A\|int, X\|Y, X_{int}\|Y_{int})
fwd(A_{int})

Pod A

Network Provider X

Network Provider Y

fwd(Y_{int})
pop(X)\npop(Y)\nfwd(dst)

Pod B

fwd(dst)

fwd(X_{int})

ToR Switch

EoP Switch

Server
Alternatives

• Other pod designs possible
  – E.g., Vswitch does the tagging at the hypervisor (if present)
  – EoP does more of the work
  – Not necessarily MPLS inside the pod

• Requirement: packet matching flowspec arrives at the egress EoP provider port with the label handle
Next Steps

• Does this make sense? 😊
  – Moving to implementation on MOC
  – Our goal is to have an architecture that does not preclude a network marketplace in the cloud

• Design client APIs, market mechanisms

• Design challenges
  – Accountability, enforcement, (price) transparency

• Extend to the wide area
Questions?

rfonseca@cs.brown.edu
dyu@cs.brown.edu
luo.mail11@imperial.ac.uk
sarianfa@cisco.com
okrieg@bu.edu
oran@cisco.com