Virtualized Access Layer

Petr Grygárek

Goals

- Integrate physical network with virtualized access layer switches
  - Hypervisor vSwitch
- Handle logical network connection of multiple (migrating) OS images hosted on physical server(s)
- Apply network policies to virtualized switches and virtualized network attachments
  - QoS, ACLs & security profiles, …
  - During VM migration, policies have to be migrated with VM
- Unified management & policies of both physical and logical network elements
  - Connects together server administrator's and network administrator's views and processes
- Network awareness of inter-VM traffic
  - Policy enforcement, statistics, packet capture, …
- Avoid extending of STP domain

Possible solution approaches

1. Implement standard network functions and APIs into software-based virtual switches
   - E.g. Cisco Nexus1000V
2. Avoid local switching and forward traffic from individual remote virtual NICs to physical switch for processing via separate logical channels
   - Dynamically create corresponding logical vEth interface on HW switch to provide configuration and feature consistency
   - Suitable when vertical traffic prevails
     - Which does not 100% apply anymore in current DC models with horizontally-scalable applications

802.1Qbg - Edge Virtual Bridging

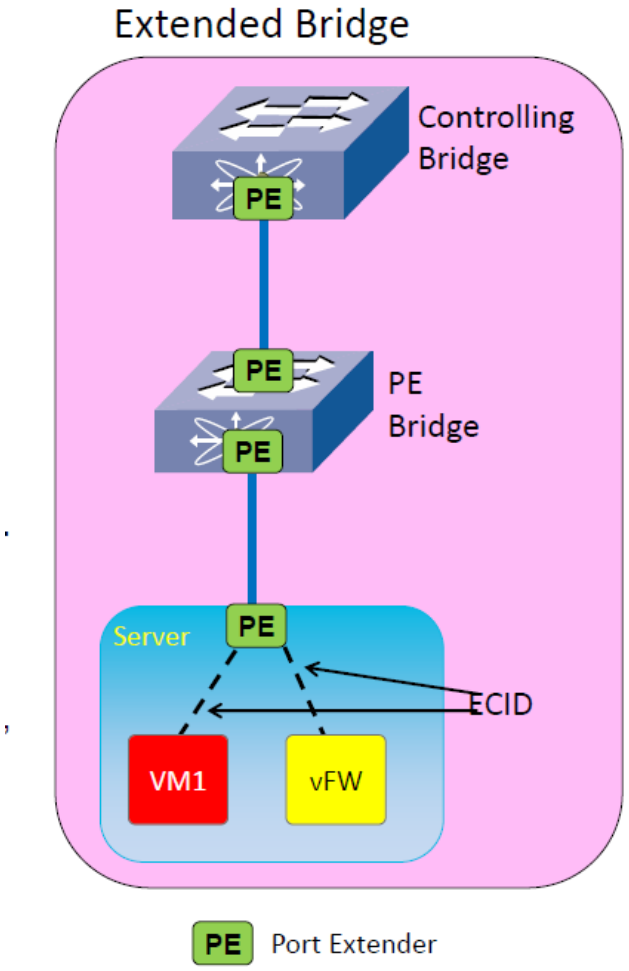
- Defines multiple technologies
- Virtual Ethernet Bridge (VEB) (roughly corresponds to VMWare vSwitch)
  - L2 inter-VM communication
  - VLAN support
- Virtual Ethernet Port Aggregator (VEPA)
  - hypervisor forwards even inter-VM traffic to external switch
    - No MAC address learning & flooding needed
    - external switch's monitoring and security tools can be enforced
    - Switching function can also be built into CNA
  - Standard Mode (tagless)
    - external switch needs to be able to forward frame back to the port which the frame came from (modified standard behaviour) – "reflective relay"
  - Multi-channel - QinQ between hypervisor and external switch
    - multiple logical attachment points for individual VNICs
    - Broadcast/multicast replication on controller switch

IEEE 802.1BR - Bridge Port Extension (1)

(originally started in 802.1qbh)

- Defines Extended bridge
- Standardized alternative to proprietary technologies like Cisco FEX
- Model of controlling (physical) bridge + Port Extender(s)
  - managed as single entity (port extenders can be understood as remote I/O cards)
  - no local switching - all traffic goes via controlling switch
  - support for remote HW-based multicast/broadcast traffic replication
- E-channel - logical channel between Extended port and corresponding virtual interface on controlling switch
- E-Tag - E-channel ID, contained in modified Ethernet frames
  - equivalent of Cisco VNTag (=slightly different format built on 802.1BR prestandard)
- Port extenders may be cascaded
  - example: controlling switch - FEX + 802.1BR-compatible server NIC (NIC virtualization) + 802.1BR-compatible hypervisors on blade servers connected to each virtualized switch – VM
  - Allows multiple network layers to be managed as single device/layer
  - Tags are NOT stacked
    - Tag- to-port mapping table still needed in Port Extenders
    - Tags are learnt together with MAC addresses on controlling SW

IEEE 802.1BR - Bridge Port Extension (2)



Port Extender Functionality

As simple as possible

- Northbound: add tag based on receiving (virtual) port & forward
- Southbound: forward based on DST VIF
- Remove E-Tag at a last hop

Bridge Port Extension Use Cases

- Physical server NIC adapter partitioning ("Adapter-FEX")
  - multiple simulated NICs presented by BIOS/PCI to OS, single attachment link to physical switch
  - tens of simulated NICs are currently supported
    - Ethernet vNICs or FibreChannel HBAs
  - dual uplink provides seamless redundancy of virtualized server NICs
    - active+standby mode, NIC teaming in OS does not need to be configured
- Virtualized physical VM-to-physical switch connection
  - fixed VEths (e.g. Redhat, Windows, VMWare ESX hypervisors)
  - floating vEths (e.g. VMWare ESX hypervisors)

VNTag/E-Tag header fields

- Presence of VNTag/Etag (4B) identified by special EtherType value (2B)
- VNTag header may be followed by 802.1q header
- Frame fields
  - Direction: indicates whether frame travels from or to remote adapter
  - Source VIF (12b)
  - Looped flag: frame looped by physical switch back to the same adapter (inter-vNIC switching)
    - Needed to avoid broadcast/multicast cycles
  - Destination VIF (12b) / VIF_List
    - if Pointer bit is specified, VIF_List is used to specify destination VIFs to replicate the frame

Port extender to controller switch Interactions

- Port extender reports number of ports to upstream switch
- upstream switch automatically creates corresponding number of tags associating each tag with single extender port

802.1q versus VNTag

- 802.1q trunk is treated by physical switch as a single port in terms of applied policies
  - Policies mostly cannot be applied per-VLAN
- All VLANs (trunk) extended to the host, server admin has to properly selects VLAN to be fed to individual VMs
  - extension of (per-VLAN) STP domains to the host

- VNTag creates virtual interfaces corresponding to vNICs that can be assigned separate policies
  - vEths treated the same way as ordinary ports by switch operating system

Cisco Nexus 5000
Static vEth Configuration Example

```
inteface veth 1
 switchport mode trunk
 bind interface Ethernet101/1/2 channel 3
```

- 101/1/2 identifies physical downlink interface (FEX-attached VNTag-capable host)
- Channel 3 identifes VIF

Cisco Nexus 5000
Dynamic vEth Configuration Example

- N5K registers itself to vSphere as vDS and reports its configured port profiles
  - Profile can be seen as port-group in vCengter
- Server administrator defines channel # and profile for each vNIC

```
vethernet autocreate

interface Ethernet1/10                  // FEX downlink
 switchport mode vntag

port-profile type vethernet MYPROFILE
 switchport mode access
 switchport access vlan 60
 port-binding dynamic
 state enabled

// created automatically
interface vethernet 23769
 bind interface ethernet 1/10 channel <# defined on server>
 inherit port-profile MYPROFILE
```

References:

- Related IEEE standards:
  - http://www.ieee802.org/1/pages/802.1br.html
  - http://www.ieee802.org/1/pages/802.1bg.html
- Comparisons of related standards (Cisco)
  - http://www.cisco.com/en/US/prod/collateral/switches/ps9441/ps9902/whitepaper_c11-620065_ps10277_Products_White_Paper.html
- Cisco FEX standards:
  - http://www.cisco.com/en/US/solutions/collateral/ns224/ns945/ns1134/qa_c67-693220.pdf
- VNTag & IEEE standards
  - http://www.ieee802.org/1/files/public/docs2009/new-pelissier-vntag-seminar-0508.pdf
- Virtual Ethernet Bridging
  - http://www.ieee802.org/1/files/public/docs2008/new-dcb-ko-VEB-0708.pdf