Software Defined Network (SDN)

University of Stuttgart – Cloud Course Fall 2013
Agenda

- Introduction
- SDN Components
- Openstack and SDN
- Example Scenario
Data Center Network – Pain Points

**Agility**
- Network and security provisioning time for a new virtual system varies from 5 days to 3 weeks (traditional network provisioning consumes 50% of the time)
- Security appliance policy configuration is complex & labor intensive → error prone

**Need for increased Bandwidth & reduced Latency**
- New workloads (Social, Mobile & Big Data) require more bandwidth & lower latency and are less predictable.
- Most data traffic remains inside the Data Center, the majority within a single rack

**Need to reduce network operational & capital costs**
- Automation of network and security tasks
- Introduction of new network capability with SDN
- Huge amount of tenants > 4K
- High reconfiguration and tenant on-boarding rate.
SDN Motivation

Today's Network Management

New Style SDN based Network Management

Direct Access to Physical Network

Network Services
- Routing API
- Traffic Engineering API
- Flow Insertion API
- Firewall API

SDN Controller & Analytics

Software Defined Control Plane

Virtualized Network

Traditional Switches
SDN Landscape

- **Open Source Implementations:**
  - **OpenFlow:** Is the physical Switch Operating System and Management API.
  - **OpenVSwitch:** Is the hypervisor virtual switch implementation supporting OpenFlow API and tunneling Protocols like GRE.
  - **SDN Controller:** Is the main entry point for System management access and configuration supporting a high level management API. E.g. OpenDaylight

- **New Network Concepts**
  - **Underlay networks:** Encapsulating today's physical networks or creating new network infrastructure based on OpenFlow technology
  - **Overlay networks:** Create a segregated per tenant view of the network.
  - **Tunneling Protocols:** Connect multiple virtual switch networks together to form a per tenant layer 2 network including segregation and individual network configuration. (VXLAN, NVGRE,...)
  - **Data and Control planes:** Divide the physical and virtual switches into the management part (Control Plane) and the high performance transport part (Data Plane) allowing to plug-in Services and API on both levels.
  - **Service Insertion Frameworks:** Provide the capability to plug-in services like (Firewall, Router, Load Balancer) into the Data Planes either on physical or virtual switch level.
Agenda

- Introduction
- SDN Components
- Openstack and SDN
- Example Scenario
SDN Components

- Network Controller providing high level management API. E.g. OpenDaylight,..
- Virtual Switches on the hypervisor. E.g. OpenVSwitch, VMware Distributed switch
- Physical switches supporting OpenFlow protocol.
- Overlay Protocols connecting the virtual switches. E.g. VXLAN, NVGRE, VMware NSX,

- Service virtual machine’s connected to the data Plane and managed via Network Controller providing high level services like:
  - Firewall
  - Load Balancing
  - VPN
  - Intrusion Detection
  - ...
OpenDaylight Network Controller

- Network applications, orchestration, and services
  - OpenDaylight APIs (REST)
  - Controller platform
    - Network service functions
    - Platform services
    - Southbound interfaces & protocols
      - OpenFlow
      - Other standard protocols (ONF, IETF, ...)
      - Vendor-specific interfaces
  - Service Abstraction Layer (SAL)
    - Extensions
    - Data plane elements (virtual switches, physical device interfaces)

Platinum: Brocade, Microsoft, Cisco, Juniper
Gold: NEC, VMware, Arista, Intel
Silver: HP, Dell, F5 Networks, Juniper, Red Hat
Virtual Switch Providing Data / Control Plane on Hypervisor

Data Plane on Hypervisor allows to plug virtual machines within the data path.
Layer 2 packets are tunneled via Layer 3 connection (UDP). Each packet has a tenant id with could be 24 / 32 or 64 bit depending on the tunnel protocol.
OpenFlow Switches

- OpenFlow switches provide together with the OpenFlow Controller a new way of configuring the network environment. Like the OpenVSwitch for the Hypervisor they provide a data and control plane allowing to:
  
  - Plug-in services at the switch layer
  - Filtering traffic based on open flow rules
  - Create a underlay network
  - Dynamically reconfigure network based on Quality of Services (QoS) and Events

- Standardized Open Source management API

- Large Vendor support
OpenFlow Routing Table

Core of an OpenFlow switch is a programmable routing table which supports Layer 2 and Layer 3 filtering

- Rule (match fields)
- Action (instructions)
- Stats (counters)

1. Forward packet to zero or more ports
2. Encapsulate and forward to controller
3. Drop packet
4. Send to normal processing pipeline
5. Modify Fields
6. Any extensions you add!

+ mask what fields to match
## OpenFlow Switch Routing Examples

### Standard switching on MAC addr

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>*</td>
<td></td>
<td>00:1f...</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>port6</td>
</tr>
</tbody>
</table>

### Switching based on application-level flow

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>port3</td>
<td>00:20..</td>
<td>00:1f..</td>
<td>0800</td>
<td>vlan1</td>
<td>1.2.3.4</td>
<td>5.6.7.8</td>
<td>4</td>
<td>17264</td>
<td>80</td>
<td>port6</td>
</tr>
</tbody>
</table>

### Firewall / filtering rules

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>22</td>
<td></td>
<td>drop</td>
</tr>
</tbody>
</table>

### IP destination-based routing

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td></td>
<td>port6</td>
</tr>
</tbody>
</table>

### MAC address switching with VLAN check (VLAN switching)

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>*</td>
<td>00:1f..</td>
<td></td>
<td>vlan1</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td></td>
<td>port6, port7</td>
</tr>
</tbody>
</table>
Tenant Level Network Overlay

- **Tenant Visible Overlay**
  - VPN / Gateway
  - Intrusion Detection
  - Firewall
  - Load Balancer
  - Virtual Machine

- **Data Plane (Hypervisor Virtual Switch)**

- **Overlay**
  - Virtual Switch
  - Overlay Controller
  - Administrator

- **Control Plane (Management API and Events)**

- **Underlay Network (OpenFlow)**
Tenant local network view

Tenant will not see how the physical network looks like. The Network services like firewall / gateway can be virtual or physical machines.
Agenda

- Introduction
- SDN Components
- Openstack and SDN
- Example Scenario
openstack as the emerging cross vendor IaaS open source cloud platform

OpenStack seeks to produce a ubiquitous Infrastructure as a Service (IaaS) open source cloud computing platform for public & private clouds.

http://www.openstack.org

Platinum Sponsors

Gold Sponsors

http://www.openstack.org
Openstack and Software Define Network

- Openstack has several components which are used in SDN.
  - Neutron is the main component which contains all the API plug points to configure networking.
  - OpenVSwitch for KVM as the virtual switch.
  - Heat as the pattern engine exposes network resource types, allowing to build network templates.
  - Horizon as the Administration UI allows:
    - Creating and configuring network resources
    - Visualizing the network setup for each tenant
    - Executing pattern via heat engine
Openstack Neutron, OpenVSwitch and SDN Controller

- Neutron Server
  - OpenVSwitch Plug-in
    - OpenVSwitch
  - Neutron Firewall Service
  - Neutron Load Balancer Service
  - Neutron VPN Service
- Neutron Firewall Service Plug-in
  - Firewall Plug-in
- Neutron Load Balancer Service Plug-in
  - Load Balancer Plug-in
- Neutron VPN Service Plug-in
  - VPN Plug-in

SDN Controller
Neutron Network Elements

- **Network**: Is the Layer 2 representation of a network
- **Port**: Is the Layer 2 connection of a VM to a network
- **Router**: Is a Layer 3 construct connecting networks
- **Subnet**: Is the Layer 3 construct with holds the IP-Address, Gateway, DNS and routes
- **Floating IP**: Is a layer 3 construct which represents an external reachable IP-Address. It can be associated to a virtual machine.
- **Load Balancer**: Is a layer 3 construct to distribute requests across a set of virtual machines. It is implemented as a service with plug point interface to vendor implementation.
- **Firewall**: Is a layer 3 construct to filter TCP/IP requests. It is implemented as a service with plug point interface to vendor implementation.
- **VPN**: Is a layer 3 construct allowing connectivity from the outside in a secure way to a network. It is implemented as a service with plug point interface to vendor implementation.
Heat Network Elements

- The heat pattern engine has 2 distinct sets of network resources:
  - CloudFormation style network definitions:
    - AWS::EC2::CustomerGateway (p. 255)
    - AWS::EC2::DHCPOptions (p. 257)
    - AWS::EC2::EIP (p. 259)
    - AWS::EC2::EIPAssociation (p. 260)
    - AWS::EC2::Instance (p. 262)
    - AWS::EC2::InternetGateway (p. 268)
    - AWS::EC2::NetworkAcl (p. 269)
    - AWS::EC2::NetworkAclEntry (p. 270)
    - AWS::EC2::NetworkInterface (p. 272)
    - AWS::EC2::NetworkInterfaceAttachment (p. 276)
    - AWS::EC2::Route (p. 277)
    - AWS::EC2::RouteTable (p. 279)
    - AWS::EC2::SecurityGroup (p. 281)
    - AWS::EC2::SecurityGroupIngress (p. 283)
    - AWS::EC2::SecurityGroupEgress (p. 286)
    - AWS::EC2::Subnet (p. 288)
    - AWS::EC2::SubnetNetworkAclAssociation (p. 290)
    - AWS::EC2::SubnetRouteTableAssociation (p. 292)
    - AWS::EC2::VPC (p. 297)
    - AWS::EC2::VPCDHCPOptionsAssociation (p. 299)
    - AWS::EC2::VPCGatewayAttachment (p. 300)
    - AWS::EC2::VPNConnection (p. 301)
    - AWS::EC2::VPNConnectionRoute (p. 303)
    - AWS::EC2::VPNGateway (p. 304)
Heat Network Elements (2)

- The heat pattern engine has 2 distinct sets of network resources:
  - Neutron style network definitions:
    - OS::Neutron::Firewall
    - OS::Neutron::FirewallPolicy
    - OS::Neutron::FirewallRule
    - OS::Neutron::FloatingIP
    - OS::Neutron::FloatingIPAssociation
    - OS::Neutron::HealthMonitor
    - OS::Neutron::IKEPolicy
    - OS::Neutron::IPsecPolicy
    - OS::Neutron::IPsecSiteConnection
    - OS::Neutron::LoadBalancer
    - OS::Neutron::Net
    - OS::Neutron::Pool
    - OS::Neutron::Port
    - OS::Neutron::Router
    - OS::Neutron::RouterGateway
    - OS::Neutron::RouterInterface
    - OS::Neutron::Subnet
    - OS::Neutron::VPNService
Agenda

- Introduction
- SDN Components
- Openstack and SDN
- Example Scenario
Example Scenario Overview Picture

Virtual Center Components

VC

V5000 Distributed Switch

Management

Smart Cloud Orchestrator (SCO)

Overlay Controller
VM

OpenFlow Controller
VM

Overlay Service VM’s

Provisioning

ESX

…

ESX

Open Flow Switches

V5000 Switch Agents

Management

Data Access

© 2013 IBM Corporation
Physical Network Elements (Switches and Hosts)
Network Underlay
Tenant Network Overlay

Domain Coke (1)
- Network HR (2)
  - Subnet (9) 192.168.3.0/24
- Network Sales (1)
  - Subnet (7) 192.168.1.0/24 & (8) 192.168.2.0/24

Domain Pepsi (2)
- Network HR (102)
  - Subnet (12) 192.168.3.0/24
- Network Sales (101)
  - Subnet (10) 192.168.1.0/24 & (11) 192.168.2.0/24
- Network HR (102)
  - Subnet (12) 192.168.3.0/24
References

- OpenFlow
- OpenVSwitch
- OpenDaylight
- Openstack Neutron, Heat and Horizon
- IBM SDN for Virtual Environments (SDN VE)
- VMware
  - vSphere 5.x
  - NSX
- Tunneling Protocols
- Software Defined Network (SDN) Book from O’Reilly
Thanks for your attention!