Using SouthBound APIs to build an SDN Solution

Dan Mihai Dumitriu
Midokura
Feb 5th, 2014
Agenda

• About Midokura
• Drivers of SDN & Network Virtualization Adoption
• SDN Architectures
• Why OpenDaylight?
• Use Cases
• Demo
• Q&A
About the company

• Founded in 2010, Midokura is a global company with offices in Tokyo, San Francisco and Barcelona

• Pioneer in network virtualization – provides software for networking using overlay approach. Pedigree derives Amazon, Cisco, VMware and Google

• Received $17M first round of funding in April 2013 from Innovation Network Corporation of Japan, NTT and NEC

• Named by CRN as amongst the top 10 networking stories of 2013 and also amongst 10 coolest startups in the world

• Adopted by service provider as well as enterprise customers

• Significant contributor to the OpenStack Networking (Neutron)

• First SDN vendor to be certified for Red Hat OpenStack environment

• Early member of the Open DayLight Project (ODP)

• Broad and deep technical partnerships with network switch vendors, software companies and solution providers

“Midokura’s distributed architecture is elegant and appears to be making strides in early adopter markets for SDN and virtual networking” – 451 research, an analyst firm

“Startups such as Midokura are taking the lead in delivering much-needed innovation to networking. The new MidoNet technology includes promising new functionality to advance network virtualization, which is critical to the success of deployment of OpenStack and other cloud solutions” – Stu Miniman, Sr. Analyst at Wikibon
Our Ecosystem

Customers

Partners

Technical Collaboration
Drivers of Adoption

• IaaS Cloud
• Public cloud
• Private cloud
SDN Architectures

• SDN doesn’t mean fully centralized controller
• It doesn’t mean remote FIB/TCAM programming
• SDN can be a combination of distributed and centralized control
  • IP (OSPF/BGP) fabric and NV overlay
The MidoNet Solution

• MidoNet is software only, highly distributed, network virtualization system

• Uses industry standards and builds highly scalable and secure virtual networks on top of an existing physical underlay.

• Using proactive overlay approach; MidoNet helps improve stability, scalability and performance of the networks.
  • Improve application deployment time – reduce time to market
  • Reduce the cost and effort to build and manage a network
  • Avoid limitations of physical network topologies
  • Improve overall user experience
MidoNet Value Prop

MidoNet reduces time to deploy applications and runs on your existing hardware

**Give your applications a race track**

Reduce network/application provisioning time from days to seconds

- Network provisioning time reduced from days to seconds

**Save existing investments**

Use your existing hardware

- Reduce CapEx and OpEx
- Improve network asset utilization
- No extra training needed for network admins

**Choice**

Multiple hypervisors
Multiple CMP
Any Hardware

- Hypervisor: KVM, ESXi, Xen, LXC
- Any CMP: OpenStack, CloudStack, Custom platforms
- Any Network Hardware

Save existing investments

- Reduce CapEx and OpEx
- Improve network asset utilization
- No extra training needed for network admins

- Network provisioning time reduced from days to seconds

- Hypervisor: KVM, ESXi, Xen, LXC
- Any CMP: OpenStack, CloudStack, Custom platforms
- Any Network Hardware
Overlay Challenges

• Limited visibility from overlays into physical networks – hard to troubleshoot networks
• Software overlays alone cannot address all workloads – physical networks need to stay
• Virtual overlays alone cannot adequately enforce policies such as QoS
• Hardware switch gateways for physical workloads
Enter OpenDaylight
The plethora of SouthBound APIs

• Many available
  • NetConf, CLI, SNMP (via MIBs), XMPP, or OpenFlow, OVSDB
  • Challenging for a company to track and implement all the SB APIs

• Solution:
  • Leverage ODP community provided SouthBound APIs
Why do I need SouthBound APIs?

- Helps to provision physical and virtual network devices
- Makes controller an intelligent device
- Allows more control to IT and Network admins
- Helps to maximize network resource utilization
- Use the right tool for the job
Architecture of MidoNet with ODP

- ODP controller runs side-by-side with MidoNet components
  - Problems of deploying virtual networks - physical networks need to be addressed since not everything is virtual
  - How to interact with hardware: CLI, NetConf, SNMP, XMPP, OpenFlow as SouthBound protocol
Overlay Networks - VXLAN

Network Virtualization provides agility, scale and rapid automation for multi-tenant environments

How?
- Decouple logical virtual networks from underlay physical fabric
- Standard IETF VXLAN overlay scheme
- With fancier control plane

Datacenter IP Fabric

Internet
Use case #1 – Virtual Physical Gateway

- Interact with Mainframe/legacy applications sitting on non virtualized hardware and non virtualized networks
  - Some workloads will remain on physical hosts for a long time for many pragmatic reasons: workloads are hard to virtualize, are on non virtualizable platforms
  - Without a vSwitch on these systems to connect to, how does one link a virtual network to these systems?
Architecture Overview – with ODL Controller and Cumulus enabled switch

Network State Database
- ODL Controller
- Analytics
- Configuration

• Horizontally Scalable
• Highly Available

Any Cloud Orchestration
VM
Agent/Switch, KVM
BGP Gateway (Cluster)
GRE/VXLAN tunnel
TCP traffic
North-bound APIs
South-bound APIs

Virtual Server
MidoNet Agent
VM
MidoNet Agent
MidoNet Agent

IP Fabric
VTEP
BGP Gateway (Cluster)

Cumulus Network Switch
L2 Network
Agent/Switch, KVM
Best of Networking and Linux Come Together

A multi-platform operating system for networking hardware that makes building, managing and automating massive-capacity next-generation datacenter networks simple.

- IPv4/IPv6 routing and bridging
- Modern datacenter & network orchestration, unifying common functions, scaling network specific ones
- Linux operations and troubleshooting
### OpenStack Instance Management

<table>
<thead>
<tr>
<th>Instance Name</th>
<th>Image Name</th>
<th>IP Address</th>
<th>Size</th>
<th>Keypair</th>
<th>Status</th>
<th>Task</th>
<th>Power State</th>
<th>Uptime</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo</td>
<td></td>
<td>102.168.0.81</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>velam0</td>
<td></td>
<td>10.0.1.3</td>
<td>206.209.300.0</td>
<td></td>
<td>Active</td>
<td>None</td>
<td>Running</td>
<td>7 hours, 30 minutes</td>
<td></td>
</tr>
<tr>
<td>velan0</td>
<td></td>
<td>10.0.1.3</td>
<td>206.209.300.0</td>
<td></td>
<td>Active</td>
<td>None</td>
<td>Running</td>
<td>6 hours, 30 minutes</td>
<td></td>
</tr>
</tbody>
</table>

**Note:**
- The screen shows a terminal window with some text, possibly related to terminal commands or output from a script.
- The diagram illustrates a UI for instance management within an OpenStack environment, showing a list of instances with various details such as name, image, IP address, size, keypair, status, task, power state, and uptime.

**Additional Details:**
- The terminal output includes logs and error messages indicating issues with certain components, such as:"DNF 4.1.x is known for its stability, speed, and ease of use, but it can also be slow and sometimes buggy. This is especially problematic when trying to install large packages or updating the system. Some users have reported issues with dependencies not being resolved properly, leading to a variety of errors. However, updates to DNF 5.x are expected to address these issues and bring a more robust, reliable experience."
Use case #2 - Troubleshooting

- If there are network problems then trouble shooting them is difficult
  - Need visibility between overlay and underlay
  - Overlay admin has no visibility in underlay

There is an opaque wall between the network admin working on the overlay network and the underlay that needs to be examined

- If Host A (with several VMs) cannot talk to H2 (with more VMs) and there are multiple routes between these hosts, how can you isolate the physical network problems if all the visibility you have is on the overlay network?
Use case #3 - App Policy

• To provide application policies, we exploit many actions, such as QoS
• Use an SB for QoS configuration
• Note that QoS is only enforceable by HW
  • Traffic classes
  • DSCP to traffic class mapping
• No standardized protocol. ODP provides all options (XMPP, CLI, NetConf, SNMP, OpenFlow, OVSDB)
  • Work with partner
What’s needed to provide complete solution

• We want to augment our solution with SouthBound APIs to interact with hardware
• OpenDaylight Project has a rich set of SBs to fill that requirement
• Real life deployment use cases dictate needs
  • Overlay
  • Underlay
  • QoS
  • Gateways
How MidoNet fits in

- We have our “controller” and SouthBound but also integrate with ODP
- the ODP Controller runs side-by-side with the MidoNet “Controller”
- We are not replacing our MidoNet controller, just complementing with ODP for extending our south bound
Thank You