Open vSwitch:
Part 2

Ben Pfaff
VMware NSBU
What is Open vSwitch?

Semi-official description:

Open vSwitch is a production quality, multilayer virtual switch licensed under the open source Apache 2.0 license. It is designed to enable massive network automation through programmatic extension, while still supporting standard management interfaces and protocols (e.g. NetFlow, sFlow, SPAN, RSPAN, CLI, LACP, 802.1ag).

The exciting parts:

• Write a program to control your network.
• Fast!
• Portable: OSes, hypervisors, CMSes, ...
Open vSwitch Hall of Fame: New Inductees

Contributors with 10 or more commits now (but not in March 2013)

Alex Wang
Alexandru Copot
Alin Serdean
Andy Zhou
Ankur Sharma
Daniele Di Proietto
Flavio Leitner
Helmut Schaa
Jean Tourrilhes
Lorand Jakab
Nithin Raju
Pavithra Ramesh
Ryan Wilson
Thomas Graf
YAMAMOTO Takashi
New Features

- Six major releases: v1.10 through v2.3.
- OpenFlow 1.1, 1.2, 1.3, some 1.4, prototypes for 1.5
- Tunnels: VXLAN, LISP, Geneve
- MPLS
- Multicast snooping
- RSTP
- TCP flags matching
New Platforms

- NetBSD
- ESX
- Windows (in progress)
Build Improvements

- More tests: 1179 → 1561
- Compiler support:
  - Clang (thread safety!)
  - MSVC 2013
- check-oftest
- check-ryu
- Travis continuous integration
- Coverity Scan
OVS Caching

ovs-vswitchd

openvswitch.ko

user

kernel

first packet

later packets

OVS Caching diagram showing the interaction between user and kernel processes, with ovs-vswitchd and openvswitch.ko as key components.
Performance, circa OVS 1.9

• 99% of traffic is not a problem:
  – Small number of long flows
  – Large number of medium-length flows.
• Tuning solves some problems.
• Real problem is large numbers of short flows:
  – Port scans
  – Peer-to-peer rendezvous servers
  – Distributed systems
  – Network monitoring applications
Megaflows (OVS v1.11)

- Most of the time the whole microflow doesn't matter, e.g. MAC learning
- Push classifier into kernel
- Hard part: userspace generates megaflows
- Exact-match cache layer
Threading (OVS v2.0)

- TCP_CRR \approx \text{ApacheBench: latency sensitive}
- Main loop is busy
- Added periodic packet processing calls
- v2.0: Packet threads
- v2.1: Revalidation, flow setup threads
  - Why is revalidation important?
- v2.3: Removed dispatcher threads
DPDK/netmap/PF_RING/…

- What are they?
- How do they help?
  - Fast and clever.
  - No baggage.
  - No ring transition.
- OVS v2.4 will support DPDK
  - Basic work was simple
  - Preliminary numbers are very good
Performance: Fairness

- Which flows get dropped?
  - Random is bad.
  - Per-tenant fairness is better.
- Per-port fairness (OVS v1.11).
- Per-destination fairness: no plans.
OVS 2014 Fall Conference

- Nov. 17 and 18 at VMware in Palo Alto
- Soliciting talks now until Oct. 6
  - User, dev, admin, research talks all welcome
- Sign up to attend now
- More information:
  - Follow link from openvswitch.org
- Free!
Questions?