Virtio Forwarder Enhancements For OVS
DPDK Full vhost Offload with SR-IOV

Hemal V. Shah, Distinguished Engineer and Architect, Broadcom Inc.
Sriharsha Basavapatna, Principal Software Engineer, Broadcom Inc.
Agenda

• vhost Full Offload with virtio Forwarder
• Problem with virtio Forwarder
• Solution Proposal
• Performance Numbers
- Involves two bridges
- Leverages SR-IOV for
  - OvS bypass and full flow offload
- **First Bridge forwards traffic between**
  - An uplink port and VF-Reps
- **Second Bridge forwards traffic between**
  - A VF & a vhost-user
  - Use simple match/action in each direction
    - “in_port=1,actions=2”
    - “in_port=2,actions=1”
  - Perform forwarding with simple lookup
  - Recommend simple rules installed before running VM
OvS Lookup Issue with virtio Forwarder

• Inefficient processing of simple flows that don’t match on header fields
  — Example of a simple flow: match on in_port and forward to another port
• Overhead of userspace DP cache/flow-table lookup
• Virtio forwarding performance is limited by unnecessary OvS overhead

Can we minimize flow lookup cost in the datapath for such flows?
Identify simple flows
- Constraints/Criteria for simple flows
- Minimal set of EM fields: in_port, dl_type, nw_frag, vlan_tci
- Flows with any other match fields don’t qualify

Manage simple flows efficiently
- Build a 64-bit mark id using minimum required EM fields
- Add the flow to a new table (simple match table) per PMD thread
- Add the flow to the normal flow table per PMD thread

Optimize datapath processing
- For the given in_port,
  - if # flows in simple match table == # flows in flow table, then the flow qualifies for a simple match
  - Build mark id and look up in simple table
  - Simple-match Hit: No miniflow-extraction/EMC/dpcls lookup; forward directly

Target to integrate it in OvS release 2.17; enhance further post-2.17

Proposal – Joint work w/ Ilya & OvS community
Performance Benefit – Preliminary Results

Perf Gain: 15-20% with simple match

Upstream-Patch-V3