Utiliser l’Intel® DPDK – Communauté dpdk.org

http://dpdk.org

Vincent.Jardin@6WIND.com
Topics

- Introduction
- Performances on Intel architecture
- Technology Overview
While Network Traffic Grows at 25% Annually….

Source: Cisco
(Worldwide data center IP traffic)
….CPU Performance Grows at only 14% Annually…. 

- CPU performance increases don’t sustain network traffic growth
And Virtualization Increases Traffic within Appliances

- Aggregate Traffic within Network Appliances
- Multiple VMs per appliance
- The "Traffic Gap"

Graph showing the increase in traffic, CPU performance, and network traffic from 2009 to 2016.
Linux Networking Kernel Doesn't Meet Scalability Needs

- Performance of Linux kernel stack doesn't scale linearly with number of cores

- Packet processing within Linux kernel can't close the network traffic gap
Traffic Growth Limits Availability of CPU Resources for Applications

- Adequate Compute Bandwidth
- Limited Compute Bandwidth
- Insufficient Compute Bandwidth

Aggregate Traffic

VMs or Applications

CPU Performance

Virtual NICs | Sockets
---|---
vSwitch | Stacks
Physical NICs | Physical NICs
Fast path Optimizes CPU Utilization

- Insufficient Compute Bandwidth
- Maximum Compute Bandwidth

VMs or Applications

<table>
<thead>
<tr>
<th>Hypervisor or OS Stack</th>
<th>Standard Linux Networking Stack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual NICs</td>
<td>Sockets</td>
</tr>
<tr>
<td>vSwitch</td>
<td>Stacks</td>
</tr>
<tr>
<td>Physical NICs</td>
<td>Physical NICs</td>
</tr>
</tbody>
</table>

minimize CPU cycles used for networking, maximizing CPU resources available for VMs or applications

Fast path-based networking stacks
- IP Forwarding, TCP, IPsec, firewall etc.

Accelerated network I/O
- PCIe, mempool etc.

Standard Linux accelerated by a fast path technology
Topics

- Introduction
- Performances on Intel architecture
- Technology Overview
Ivy Bridge Platform Description

- **Intel Crown Pass platform**
  - Dual Ivy Bridge processors (each with 12 cores)
  - 3.5GHz CPU speed
  - 32GB RAM
  - 22 x 10G interfaces (327Mpps)

- **Traffic generator**
  - Up to 24 x 10Gbps interfaces using daisy chain
  - Up to 357 Mpps.
IP Forwarding using http://dpdk.org

Test Results

- Fast path IP forwarding performance
  - 14.24 Mpps per core
  - 24 Mpps per core with l2switch
  - Up to 313.31 Mpps with 22 cores

- Performance scales linearly with the number of cores configured to run the fast path.

- Performance is independent of frame size.
IP Forwarding using http://dpdk.org

Test Results

- Fast path IP forwarding performance
  - 14.24 Mpps per core
  - 24 Mpps per core with l2switch
  - Up to 313.31 Mpps with 22 cores

- Performance scales linearly with the number of cores configured to run the fast path.

- Performance is independent of frame size.
IPsec using http://dpdk.org

Test Platform

AES128-HMAC-SHA1 for all the measurements

3 measurements:

• Software crypto

• PCIe crypto using Intel Cave Creek

• PCIe crypto using Cavium Nitrox
Software IPsec Test Results

- **IPsec performance**
  - 5.39 Gbps per core for 1420B packets
  - Up to 73.01 Gbps using 14 cores / 28 threads

- Performance scales linearly with the number of cores configured to run the fast path
Software IPsec Test Results

- IPsec performance using DPDK AES-NI/AVX software crypto
  - 1.81 Gbps per core for 64B packets
  - 5.39 Gbps per core for 1420B packets
  - Up to 73.01 Gbps using 14 cores / 28 threads

- Performance scales linearly with the number of cores configured to run the fast path
IPsec with PCIe Intel Cave Creek Test Results

- IPsec using Quick Assist DPDK addon
  - 3.52 Gbps per engine for 1420B packets
  - Up to 40 Gbps (platform limit) using 16 engines

- Performance scales linearly with the number of engines configured to process IPsec transformation
IPsec with PCIe Intel Cave Creek Test Results

- IPsec using Quick Assist DPDK addon
  - 13.56 Gbps with 16 engines for 64B packets
  - Up to 40 Gbps (platform limit) with 16 engines for 1420B packets

![6WINDGate IPsec performance: Cave Creek](chart.png)
IPsec with PCIe Cavium Nitrox

Test Results

- IPsec performance using Cavium Nitrox DPDK add-on
- Up to 20.23 Gbps for 1420 bytes
Using a fast path in a Virtual Environment: Complete Solution

- **Accelerate packet processing in the hypervisor**
  - Leverages PMDs for physical NICs
  - Accelerates virtual switching / routing thanks to the fast path (OVS, V(x)LAN, (NV)GRE + other protocols)
  - Enables high performance communication with the VMs using a vNIC PMD

- **Accelerate packet processing in the VMs**
  - Enables high performance communication with the virtual switch using a vNIC PMD
  - Supports vNIC netdevice if Intel® DPDK is not required in the VM
  - Accelerates packet processing thanks to the fast path (NFV, TCP)

- **In addition, communication with standard VMs using standard vNICs**
OVS Acceleration: Performance

- **OVS L2 switching performance**
  - 6.8 Mpps per core
  - Up to 67.8 Mpps using 10 cores (20 threads)

- Performance scales linearly with the number of cores configured to run the fast path.

- Performance is independent of frame size.
Topics

- Introduction
- Performances on Intel architecture
- Technology Overview
Fast Path Architecture

- Synchronization module
- Control Plane
- Continuous synchronization
- Local info
- Kernel slow path
- Exception packet
- Fast path packet
- DPDK
Your Three Options for Obtaining Intel® DPDK

1. From Intel
   - As direct Intel customer
   - Latest version of Intel® DPDK
   - www.intel.com/go/dpdk

2. From dpdk.org
   - Latest version of Intel DPDK
   - Community patches
   - Community support
   - Some 6WIND-developed features

3. From 3rd parties
   - Maintained version
   - Latest version of DPDK
   - Feature and performance enhancements
   - dpdk.org patches
   - Full technical support and maintenance
   - Optionally integrated with a fast path stack
6WIND's Enhancements on http://dpdk.org

Developed by 6WIND, available from dpdk.org
Implement a fast path on http://dpdk.org

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>QoS</td>
<td>IPv4 / IPv6 Multicast</td>
<td>IP Reassembly</td>
<td>GTP-U</td>
<td></td>
<td>L2TP, PPPoE BRAS</td>
<td>TCP / UDP Termination</td>
</tr>
<tr>
<td>IPv4 / IPv6 Filtering</td>
<td>MPLS / VPLS Encapsulation</td>
<td>NAT</td>
<td>IPsec SVTI</td>
<td>Open vSwitch (OVS) Acceleration</td>
<td>Extended Fast Path (1CP + xFP)</td>
<td></td>
</tr>
</tbody>
</table>

RX/TX plug-ins (load balancers, QoS, custom modules)

librte_mbuf ➔ fast path buf  rte_timers ➔ timers  DPVI  fpu RPC

librte_ether (rte_eth_dev_*)

librte_crypto  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so

dpdk.org  librte_eal/linux userland  librte_mempool  librte_mempool  librte_mempool  librte_mempool  librte_mempool  librte_mempool  librte_mempool

librte_pmd_ring  librte_pmd_e1000  librte_pmd_ixgbe  librte_pmd_pcap  librte_pmd_ring  librte_pmd_virtio  librte_pmd_virtio  librte_pmd_virtio  librte_pmd_virtio

librte_pmd_vmxnet3.so  librte_pmd_mlx4.so  librte_pmd_CONFID.so  librte_pmdCumhur.so  librte_pmdCONFID.so  librte_pmdｄｅｎａｄｅｍ４．so  librte_pmdｄｅｎａｄｅｍ４．so  librte_pmdｄｅｎａｄｅｍ４．so  librte_pmdｄｅｎａｄｅｍ４．so

librte_power  librte_sched  librte_power  librte_sched  librte_power  librte_sched  librte_power  librte_sched  librte_power

librte_cmdline  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so

rte_mbuf ➔ fast path buf  rte_timers ➔ timers  DPVI  fpu RPC

librte_ether (rte_eth_dev_*)

librte_crypto  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so  librte_pmd_multibuffer.so

dpdk.org  librte_eal/linux userland  librte_mempool  librte_mempool  librte_mempool  librte_mempool  librte_mempool  librte_mempool  librte_mempool

librte_pmd_ring  librte_pmd_e1000  librte_pmd_ixgbe  librte_pmd_pcap  librte_pmd_ring  librte_pmd_virtio  librte_pmd_virtio  librte_pmd_virtio  librte_pmd_virtio

librte_pmd_vmxnet3.so  librte_pmd_mlx4.so  librte_pmd_CONFID.so  librte_pmdCumhur.so  librte_pmdCONFID.so  librte_pmdｄｅｎａｄｅｍ４．so  librte_pmdｄｅｎａｄｅｍ４．so  librte_pmdｄｅｎａｄｅｍ４．so  librte_pmdｄｅｎａｄｅｍ４．so

librte_power  librte_sched  librte_power  librte_sched  librte_power  librte_sched  librte_power  librte_sched  librte_power

librte_cmdline  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so  librte_pmd_quickassist.so
fast path =
- Intel® DPDK from http://dpdk.org
- + a software stack

Turbo boost your Linux, your vSwitch, your networking solutions