

Download PDF

访问提供者 :  
Hytera Communication Corp.Ltd  
登出

通过利用SR-IOV和DPDK数据包处理加速来增强VNF性能

<< Results

8 作者 (5)

Michail-Alexandros Kourtis ; Georgios Xilouris ; Vincenzo Riccobene ; Michael... 查看所有作者

8 纸引文

862 充分文字视图

Export to CollaborateC Alerts

Manage Content Alerts Add to Citation Alerts

更喜欢这个

结合强制访问控制和虚拟机内省的虚拟化安全  
2014 IEEE / ACM第7届公用事业和云计算国际会议  
发布时间 : 2014年

克服大型vCPU虚拟机的虚拟化开销  
2018年IEEE第26届计算机和电信系统建模, 分析和仿真国际研讨会 ( MASCOTS )  
发布时间 : 2018年

查看更多

摘要

网络功能虚拟化 ( NFV ) 的主要目标是将物理网络功能迁移到云计算中虚拟机 ( VM ) 上运行的软件版本... 查看更多

元数据

摘要 : 网络功能虚拟化 ( NFV ) 的主要目标是将物理网络功能迁移到在云计算环境中的虚拟机 ( VM ) 上运行的软件版本。虚拟化技术的快速发展使得高速网络连接和线速数据包处理在虚拟化基础架构环境中可行。选择计算密集型虚拟化网络功能 ( VNF ) 形式的网络流量的深度包检测 ( DPI ) 作为代表性用例。使用DPI用例来演示使用支持SR-IOV的设备和DPDK来支持高性能虚拟网络功能 ( VNF ) 部署的好处。已经使用LibPCAP , SR-IOV和DPDK对VNF版本进行了性能评估。

发表于 : 2015年IEEE网络功能虚拟化和软件定义网络会议 ( NFV-SDN )

会议日期 : 2015年11月18日至21日

INSPEC登录号 : 15723680

IEEE Xplore日期 : 2016年1月21日

DOI : 10.1109 / NFV-SDN.2015.7387409

电子ISBN : 978-1-4673-6884-1

出版商 : IEEE

会议地点 : 美国加利福尼亚州旧金山

Citation Map

1. PCI Special Interest Group, [online] Available: http://www.pcisig.com/home. Show Context Google Scholar

2. DPDK: Data Plane Development Kit, 2015, [online] Available: http://dpdk.org/. Show Context Google Scholar

3. K. Salah, A. Qahtan, "Implementation and experimental performance evaluation of a hybrid interrupt-handling scheme", Computer Communications, vol. 32, no. 1, pp. 179-188, January 2009. Show Context CrossRef Google Scholar

作者

数据

参考

引文

关键词

度量

More Like This

请参阅本文中提到的技术专利的顶级组织

ORGANIZATION 4  
ORGANIZATION 3  
ORGANIZATION 2  
ORGANIZATION 1

单击以展开

Provided by: Innovation PLUS  
POWERED BY IEEE AND IF.COM  
A PATENT SEARCH AND ANALYTICS TOOL

4. "Wind River Content Inspection Engine", [online] Available:  
<http://www.windriver.com/products/product-overviews/POWind-River-Content-Inspection-Engine.pdf>.  
[Show Context](#)   [Google Scholar](#)

---

5. "Protocol and Application Classification with Metadata Extraction", [online] Available: <http://www.ipoque.com/en/products/pace>.  
[Show Context](#)   [Google Scholar](#)

---

6. [online] Available: <http://www.gosmos.com>.  
[Show Context](#)

---

7. L. Deri, M. Martinelli, T. Bujlow, A. Cardigliano, "nDPI: Open-Source High-Speed Deep Packet Inspection", *International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 617-622, 4–8th Aug. 2014.  
[Show Context](#)   [Google Scholar](#)

---

8. T. Bujlow, V. Carela-Español, P. Barlet-Ros, Extended Independent Comparison of Popular Deep Packet Inspection (DPI) Tools for Traffic Classification, Universitat Politècnica de Catalunya, 2014.  
[Show Context](#)   [Google Scholar](#)

---

9. [online] Available:  
<https://www.hastexo.com/system/files/neutronpacketflows-notes-handout.pdf>.  
[Show Context](#)

---

10. L. Rizzo, "Netmap: a novel framework for fast packet I/O", *USENIX Annual Technical Conference*, April 2012.  
[Show Context](#)   [Google Scholar](#)

---

11. "PF\_RING ZC", [online] Available:  
[http://www.ntop.org/products/pf\\_ring/pf\\_ring-zc-zero-copy/](http://www.ntop.org/products/pf_ring/pf_ring-zc-zero-copy/).  
[Show Context](#)   [Google Scholar](#)

---

12. *Pktgen version 2.7.7 using DPDK-1.7.1*, [online] Available:  
<https://github.com/Pktgen/Pktgen-DPDK/>.  
[Show Context](#)   [Google Scholar](#)

Advertisement

## Contents

### 第一节 介绍

软件定义网络（SDN）的激增和网络功能虚拟化（NFV）的推出引起了电信领域的极大关注，并承诺采用新的网络设计方法。NFV的网络模型专注于在虚拟化基础架构上运行的基于软件的网络功能的实施。此概念使多个虚拟化网络功能可以同时在一个大容量服务器上运行。因此，运营商具有动态启动，重新配置和终止VNF的能力，从而获得了显著的灵活性和灵活性。但是，为了实现这些期望，必须考虑大量的考虑因素。在传统的网络设备中，软件组件针对其定制硬件平台进行了高度优化。

当VNF部署在云基础架构（例如OpenStack）中托管的虚拟机（VM）上时，单个服务器设备预期要处理的网络流量将显着增加。当多个VNF同时运行并且每个VNF每秒处理大量数据包时，可能会产生重大的性能挑战。系统的体系结构功能（如内存访问，任务和资源分配，网络I/O等）可对性能产生重大影响。

由于I/O性能在云基础架构中至关重要，因此需要进行虚拟化优化，以便最大限度地利用计算机系统资源。单根I/O虚拟化（SR-IOV）[1]是PCI-SIG发布的规范，它定义了减少虚拟机管理程序与VM交互的硬件增强，以提高其数据处理性能。支持SR-IOV的设备能够产生PCI功能的各

种“轻”实例，称为虚拟功能（VF）。每个VF都可以固定到VM，允许直接访问其物理资源。利用此功能，多个VM可以共享相同的物理PCI设备资源，从而实现高性能，而不会产生大量额外的CPU开销。由于SR-IOV提供对系统硬件资源的直接访问和控制，因此可以有效分配低级网络资源。

为了在网络和计算领域充分利用系统资源，同时增强和促进密集网络应用的实施，英特尔开发了数据平面开发套件（英特尔®DPDK）[2]。DPDK包含一组库，这些库通过访问系统的网络接口卡（NIC）来支持网络功能的高效实现。DPDK为网络功能开发人员提供了一套用于构建高速数据平面应用程序的工具。DPDK在轮询模式下进行数据包处理，而不是默认的中断模式。轮询模式操作采用忙等待技术，不断检查网络接口的状态变化。这减轻了数据包处理的中断，因为它绕过了内核，有效地消耗了CPU周期，从而导致数据包吞吐量增加[3]。使用DPDK网络数据包入口和出口与标准Linux内核网络堆栈相比更快，因为用户空间支持应用程序，从而绕过内核网络堆栈瓶颈。

在本文中，选择了一个网络 and 计算密集型用例，以展示使用支持SR-IOV的设备和DPDK支持高性能VNF部署的好处。选择网络流量的深度包检测（DPI）作为可行且合适的用例。所需的数据包分析和流信息进程消耗大量的计算资源，因为所有网络流和它们的大多数数据包需要在协议报头的深处处理，在许多情况下在有效负载级别，以提供准确和精确的网络数据包识别交通。DPI主要用于识别各种安全性或网络管理目的的网络流量配置文件。DPI分析从第2层到第7层的IP流量，包括报头和数据协议结构以及数据包的有效负载 的消息。此信息用于标识各种应用程序协议和流量。该信息的收集可用于提供被监视网络的流量分类。WindRiver等公司存在各种商业DPI解决方案[4]，PACE [5]和Qosmos [6]。Qosmos DPI提供基于Qosmos旗舰产品ixEngine的虚拟化解决方案，该产品已成为电信和企业解决方案开发商事实上的行业标准DPI引擎。另一种免费的开源解决方案是nDPI [7]。NDPI是流行的OpenDPI库的ntop维护超集。与其他开源DPI库和思科NBAR相比，NDPI在[8]中进行的广泛流量识别实验中表现更好。

本文中描述的DPI解决方案是使用nDPI库构建的。该方法仅使用来自每个流的指示性的少量初始分组，以便识别有效载荷内容，并且不检查每个分组。在这方面，它遵循基于分组的每流状态（PBFS）。此方法使用表根据为每个流维护的5元组（源地址，目标地址，源端口，目标端口和传输协议）跟踪每个会话。

本文的结构如下：第二部分概述了问题陈述，第三部分介绍了详细的架构和解决方案实施，第四部分评估并提出了拟议解决方案的绩效结果。第五部分扩展了有关当前虚拟化基础架构中自动VNF部署和增强平台感知的讨论。最后，第六节总结了论文并描述了这项研究工作的后续步骤。

## 第二节 问题陈述

诸如OpenStack之类的云基础架构解决方案运行在当前操作系统提供的各种网络组件之上，例如路由，过滤，流重建等。这些功能会产生大量开销，导致性能损失，因为每个数据包都通过操作系统内核并不能直接访问用户空间。Linux网络堆栈通常用作云网络解决方案的基础，其主要目标是为全功能操作系统提供通用网络堆栈。因此，标准Linux网络堆栈无法扩展到软件路由器应用程序所需的性能级别。一般来说，OpenStack的网络服务Neutron基本上是基于Open vSwitch（OVS）实例，它包括各种本机OVS虚拟端口，TAP虚拟网络内核接口和连接它们的内核桥接器，用于每个VM实例。这种多桥虚拟接口设置引入了数据包转发性能的大幅延迟，因为每个数据包必须经过并由内核多次处理[9]，在它到达最终目的地之前。使用这种形式的架构设计，操作系统的内核性能很快就会成为瓶颈。

## 第三节 描述和实施

两种设置配置用于流量分类VNF的性能测试。两个测试部署均使用采用Intel®Xeon®E5-2620v3 - 2.40GHz CPU的服务器执行。每台服务器都有一个双端口10Gbit英特尔®以太网聚合网络适配器X520-T2。在两种配置中，一台服务器用作流量生成器，第二台服务器托管DPI应用程序。第一个测试在物理层执行，意味着DPI应用程序直接使用物理网卡执行，而在第二个测试中，DPI应用程序通过KVM管理程序部署为VM，数据包通过SR-IOV快速路径到达。在两个性能测试中，使用LibPCAP评估两个版本的DPI应用程序，使用DPDK评估一个版本。第一个配置的目的是为DPDK增强型DPI应用程序的基准性能和基准性能设置基准。此外，该测试评估了虚拟化对性能的成本开销。各种数据包加速候选框架Netmap [10]，PF\_RING [11]，以及提供零拷贝，内核旁路和支持多队列功能的DPDK，选择英特尔®DPDK来实现VNF的加速版本，因为它提供了高级用户级功能，例如具有增强功能的多核框架NUMA感知和跨不同核心的数据包操作库。DPDK库还提供两种执行模型：一种流水线模型，其中一个核心具有将数据包从接收NIC传输到另一个核心进行处理的能力，以及运行到完成模型，其中数据包分布在所有核心之间以加速处理。数据包调度的特性和不同的执行模型使DPDK框架不仅仅是I/O库和I/O库。

A. 带有SR-IOV和DPDK的VM测试台

在图1和2中，示出了两种配置设置的详细概述。图1显示了基线设置，其中未进行任何其他修改以增强DPI应用程序的性能。在物理NIC和虚拟NIC驱动程序层中，数据包由Linux内核网络堆栈处理。两个测试（物理和虚拟化）中的DPI应用程序都使用LibPCAP来读取和分析收到的网络流量。

图2显示了增强的体系结构，其中已在主机服务器的物理网卡上启用了SR-IOV，相应的VF驱动程序已连接到DPI虚拟机，最后虚拟网卡加载了DPDK驱动程序以实现更快的NIC-用户空间通信。在这种情况下，DPI应用程序在物理和虚拟化实验中读取和处理使用DPDK框架接收的数据包。

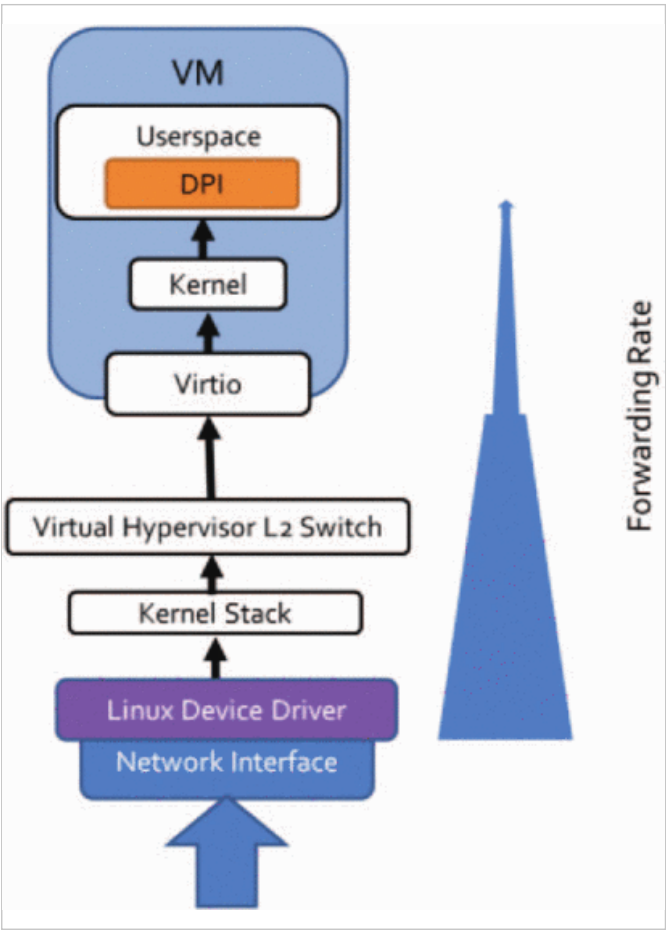


图. 1。  
标准设置

图2显示了主机服务器的物理网卡上的SR-IOV支持。相应的VF驱动程序连接到DPI虚拟机。虚拟NIC加载了DPDK驱动程序，以便更快地进行

NIC用户空间通信。在此配置中，DPI应用程序在物理和虚拟化实验中读取和处理使用DPDK框架接收的数据包。

PDF

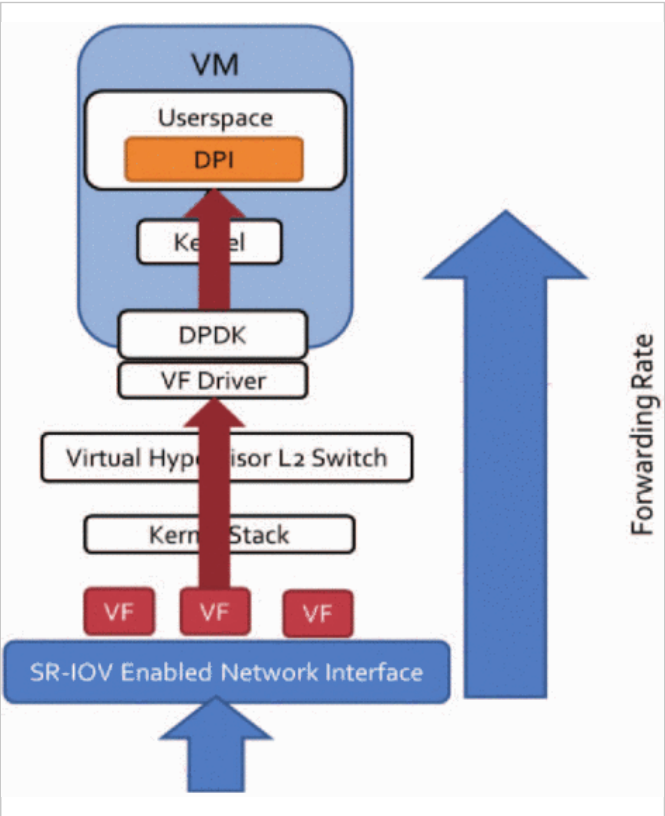


图2. 在VM上使用启用了SR-IOV的端口和DPDK进行设置。

这些实验的主要目标是评估和量化SR-IOV和DPDK支持的VNF与标准Linux网络堆栈相比的性能改进。此外，评估测试的目的是量化计算密集型网络功能（例如DPI）中的虚拟化成本。

对于DPI VNF的评估和基准测试，使用了DPDK PktGen [12]流量生成器，该流量生成器建立在DPDK快速数据包处理框架之上，以实现线速流量生成。即使使用64字节帧，PktGen也能够产生10Gbps的流量速率。出于实验的目的，基于数据文件的PCAP流量重放被使用并以10Gbps速率重传。在NCSRD中捕获的实际流量跟踪用于生成PCAP文件。

图3显示了数据包大小的分布。可以清楚地看到，大多数数据包属于1280-1518字节的大小组，第二个最常用的大小组是40-79字节类别。对PCAP文件的分析表明DPI暴露于丰富的流量类型和协议集，这确保了真实的测试场景。总共捕获的PCAP流量文件由2343个唯一流和28个不同的应用协议组成。

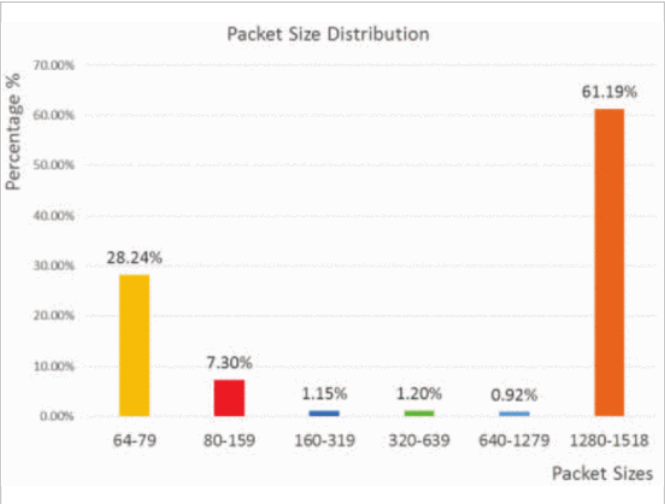


图3. 包大小分布图

**B. NDPI与DPDK的整合**

nDPI的标准实现建立在数据包处理库LibPCAP之上。当接口接收数据包时，nDPI可以从pcap头中提取数据包的时间戳。然后删除pcap标头以访问和读取数据包的内容。DPDK使用其自己的基元类型来存储和访问数据，但是为了使接收的数据包与DPDK兼容，需要进行转换。转换包括网络类型，例如IP头类型，以及原始类型，例如unsigned char类型，以便充分利用DPDK框架的用户级功能。

除了类型转换和移植到DPDK之外，还进行了自上而下的nDPI数据包处理重新设计，以便将DPDK平面带到nDPI识别的层。标准nDPI库主要用于在访问网络分组数据方面与LibPCAP一起正常运行。对nDPI端进行了各种修改，以促进与DPDK框架的集成。

**C. VM互连**

VNF正常运行需要服务功能链（SFC），流量转发和VM间通信。然而，将VNF功能集成到OpenStack的网络环境中，更具体地说，与Neutron服务的功能集成是非常重要的，并且仍然有待实施，因为Neutron目前提供有限的功能来支持任意流量转向。未来对这项工作的增强将包括将当前工作集成到用于虚拟化网络基础设施的自动，灵活和高效的网络设备中，因为DPDK和SR-IOV目前需要各种复杂的系统配置才能正常运行并提供改进的性能。

为了支持直接流量转发，即一个虚拟网络功能组件（VNFC）的虚拟网络接口直接连接到另一个VNFC的虚拟网络接口，需要修改Neutron的OVS。VNFC的每个虚拟网络接口都反映在一个TAP虚拟网络内核设备，Neutron OVS上的虚拟端口以及连接它们的虚拟网桥上。数据包通过Linux内核从VNFC传输到Neutron的OVS。需要关闭和删除两个VNFC的虚拟内核桥接，然后需要在Neutron OVS上应用OVSDB规则。此规则定义相应VNFC的OVS端口之间的所有转发策略。

**第四节  
实验结果**

本节介绍DPIK和LibPCAP版本的DPI在物理和虚拟化环境中的比较测试结果。两个实验评估的流量由流量发生器以线性方式从1到100%生成，每秒增量为1%。每秒从VNF收集流量统计数据，并进行后处理以进行性能评估。

图4显示了在裸机情况下基于LibPCAP的DPI部署与VNF的DPDK加速版本的数据包处理性能的比较。结果清楚地表明，当DPDK用于加速数据包处理时，DPI的性能得到显著提高。LibPCAP版本表现出大约1 Gbps的饱和度和。与DPDK加速版本的近似线速性能相比，该吞吐量较差。

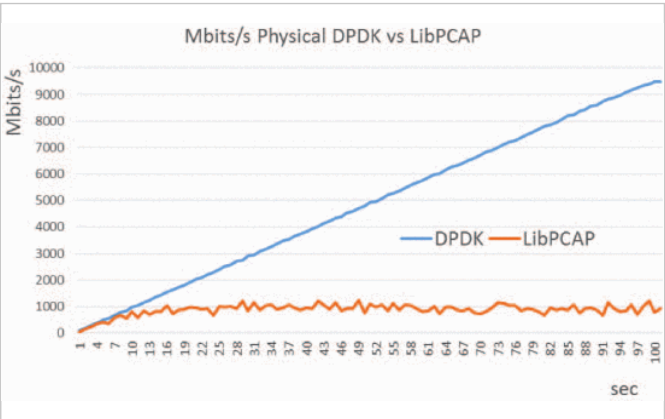


图4. 物理测试平台DPDK与LibPcap。

第二组实验侧重于确定在部署为VM时将SR-IOV和DPDK组合在DPI性能上的效果。使用高达10Gbps的线性扩展网络流量负载来对DPI VM的



LibPCAP和SR-IOV / DPDK版本进行压力测试。

Down

PD5 如图5所示，SR-IOV / DPDK版本实现了物理DPDK测试数据包传输性能的大约81%。LibPCAP版本在1Gbps时显示饱和效果，与DPDK版本相比，吞吐量降低了87.5%。结果还表明，与相应的物理测试相比，DPDK在虚拟化场景中的性能下降了大约19%。这种性能下降可以解释为DPDK在虚拟化环境中运行，并不真正绑定CPU内核，因此不会真正执行线程/CPU隔离。

结果表明虚拟化DPI解决方案具有良好的性能，并且在DPDK使用时有明显的改进。物理和虚拟解决方案之间的性能差距表明需要进一步优化，以使VNF实现接近相应物理DPI解决方案即线路速率的性能。此外，很明显，尽管使用了SR-IOV，网络内核堆栈仍然是数据包处理路径的瓶颈。

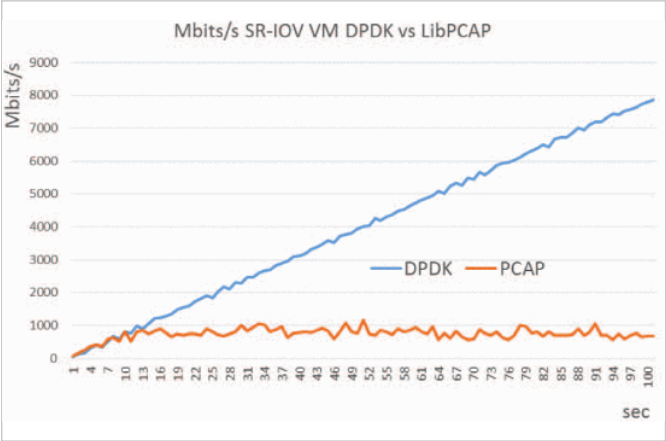


图5。SR-IOV测试平台DPDK与LibPcap之间的虚拟机。

### 第五节 关于自动部署的注意事项

上一节提供的证据表明，数据包加速技术解决方案及其适当的集成可以为运行网络功能的实际物理设备（例如中间盒）的虚拟化实例提供可比较的性能。但是，用于托管VNF的当前虚拟化基础架构环境带来了许多管理和配置挑战。例如，OpenStack已将自身确立为实施和管理云基础架构的事实上的开源解决方案，因此需要进行大量的手动配置，以便在VNF之间以正确的方式正确创建所需的服务链链接。

但是，假设已经解决了所有前置和后置配置注意事项，则VNF实现不应与托管平台功能无关。从本质上讲，在开发VNF期间以及在现实环境中部署之前的测试期间，需要仔细考虑DPDK等功能。此外，提高OpenStack和其他云框架的平台意识的努力正在获得动力。提高平台意识将支持将VNF适当地自动映射到具有所需平台功能的最合适的计算节点。

由于DPDK在VM内或裸机上的操作模式，依赖于CPU使用率监控的扩展机制无法触发正确的决策。这是因为当在VNF中使用DPDK时，无论网络接口上的流量负载（虚拟或非虚拟），CPU核心的利用率都达到100%的上限。此外，使用DPDK时，Linux内核堆栈不再识别NIC。因此，NIC没有TCP / IP堆栈，因为内核被绕过。此外，启用DPDK的NIC无法响应ICMP请求，因此无法从第3层网络角度发现。这为启用DPDK的VNF的开发增加了一些额外的开销。这些是可接受的缺点，以便实现显着改进的性能，但是OpenS大头钉框架中固有的自动缩放机制甚至手动缩放机制将无法正确扩展。解决方案是使用VNF内部的VNF特定监视信息，并在需要时创建用于扩展VNF的新规则和方法。

基于先前的考虑，适当选择VNF的数据路径加速功能是在基础设施和管理灵活性方面的权衡。这种选择会影响NFV链中不同程度的各种参与者。综上所述：

- 功能开发人员

- Down PDF
- VNF代码开发（代码应该适当考虑以支持DPDK）。此要求的风险有限，因为DPDK提供了大量方法库，以便为多个网络功能提供集成和移植服务。
  - VNF开发环境：由于VNF运行的操作环境的异构性，在开发过程中考虑增强的平台功能可能具有挑战性。
  - 基础设施提供商
    - 供应商锁定和成本。采购提供增强功能的设备。例如，需要专用PCIe协处理卡（例如FPGA）来提供特定数据包处理能力的VNF可能会导致一定程度的供应商依赖性。但是，由于云生态系统主要基于技术标准，因此与其他技术领域相比，供应商锁定不再是一个问题。
    - 向Orchestrator提供平台信息，以便在NFVI-Point of Presence（PoP）计算节点之间实现应用类型与资源分配的最佳映射。
  - 服务提供商
    - 在Orchestration级别获得增强平台功能（即基础架构存储库，服务映射模块）的支持，以便将VNF所需的资源和功能映射到底层基础架构。

第六节

结论和未来的工作

已经描述了支持高速网络分组处理的虚拟化DPI。各种实例中的VNF已经过基准测试，以评估其性能。DPI VNF使用LibPCAP和Intel的DPDK框架实现。还使用了SR-IOV框架，以便在虚拟化环境中最大化分组吞吐量。此外，还提出了有关在云环境中自动部署增强型VNF的各种注意事项和讨论。已经描述和分析了各种限制和发展限制。性能评估结果表明，与使用Linux内核网络堆栈的数据包处理相比，使用SR-IOV和DPDK可以实现更高的性能。需要额外的工作来通过启用SR-IOV / DPDK的DPI版本和10Gbps线速性能来缩小剩余间隙。这项工作的下一步将侧重于NUMA和核心固定以及巨大页面大小的优化，以实现DPI的近线速率性能。

致谢

这项工作是在欧盟FP7 T-NOVA项目的信息通信技术下进行的，该项目由欧盟委员会根据619520的拨款部分资助。

作者

▼

数据

▼

参考

^

引文图

1. PCI特别兴趣小组, [在线]可用: [http : //www.pcisig.com/home](http://www.pcisig.com/home)。  
显示上下文 Google学术搜索

2. DPDK : 数据平面开发工具包, 2015, [在线]可用: [http : //dpdk.org/](http://dpdk.org/)。  
显示上下文 Google学术搜索

3. K. Salah , A. Qahtan , “混合中断处理方案的实施和实验性能评估” , 计算机通信, 第一卷。32 , 不。1 , pp.179-188 , 2009年1月。  
显示上下文 CrossRef 谷歌学术

4. “风河内容检测引擎” , [在线]可用 :

<https://ieeexplore.ieee.org/document/7387409/references#references>

8/10



[http : //www.windriver.com/products/product-overviews/POWind-River-Content-Inspection-Engine.pdf](http://www.windriver.com/products/product-overviews/POWind-River-Content-Inspection-Engine.pdf).

Download PDF显示上下文 Google学术搜索

5. “使用元数据提取的协议和应用程序分类” , [在线]可用 :  
[http : //www.ipoque.com/en/products/pace](http://www.ipoque.com/en/products/pace)。  
显示上下文 Google学术搜索

6. [在线]可用 : [http : //www.gosmos.com](http://www.gosmos.com)。  
显示上下文

7. L. Deri , M. Martinelli , T. Bujlow , A. Cardigliano , “nDPI : 开源高速深度包检测” , *国际无线通信和移动计算会议 ( IWCNC )* , 第617-622页 , 第4页 - 2014年8月8日。  
显示上下文 Google学术搜索

8. T. Bujlow , V. Carela-Español , P. Barlet-Ros , 用于交通分类的流行深度包检测 ( DPI ) 工具的扩展独立比较 , *Universitat Politècnica de Catalunya* , 2014。  
显示上下文 Google学术搜索

9. [在线]可用 : [https : //www.hastexo.com/system/files/neutronpacketflows-notes-handout.pdf](https://www.hastexo.com/system/files/neutronpacketflows-notes-handout.pdf)。  
显示上下文

10. L. Rizzo , “Netmap : 一种新的快速数据包I / O框架” , *USENIX年度技术会议* , 2012年4月。  
显示上下文 Google学术搜索

11. “PF\_RING ZC” , [在线]可用 :  
[http : //www.ntop.org/products/pf\\_ring/pf\\_ring-zc-zero-copy/](http://www.ntop.org/products/pf_ring/pf_ring-zc-zero-copy/)。  
显示上下文 Google学术搜索

12. *Pktgen*版本2.7.7使用*DPDK-1.7.1* , [在线]可用 :  
[https : //github.com/Pktgen/Pktgen-DPDK/](https://github.com/Pktgen/Pktgen-DPDK/)。  
显示上下文 Google学术搜索

引文	▼
关键词	▼
度量	▼

IEEE Account	▼
Profile Information	▼
Purchase Details	▼
Need Help?	▼
Other	▼

A not-for-profit organization, IEEE is the world's largest technical professional organization dedicated to advancing technology for the benefit of humanity.  
© Copyright 2019 IEEE - All rights reserved. Use of this web site signifies your agreement to the terms and conditions.

US & Canada: +1 800 678 4333  
Worldwide: +1 732 981 0060

IEEE帐户	购买细节	档案信息	需要帮忙？
» 更改用户名/密码	» 付款方式	» 通讯首选项	» 美国和加拿大： +1 800 678 4333
» 更新地址	» 订单历史	» 职业与教育	» 全球： +1 732 981 0060
	» 查看购买的文档	» 技术兴趣	» 联系与支持

