



基于 Docker 的 DPI 虚拟化部署

潘洁, 薛翌

(中国移动通信集团设计院有限公司, 北京 100080)

摘要: DPI 是运营商网络中用于流量识别和处理的专用设备, 目前被大量部署于现网中。但传统的 DPI 部署方式带来了多个 DPI 系统并存、功能重复、维护管理复杂等问题, 不利于 DPI 应用的大量标准化快速部署。随着 NFV 技术的发展, 对 DPI 进行虚拟化部署成为解决此问题的一种趋势。Docker 是一种容器级的虚拟化技术, 可在其基础上进行 DPI 部署。阐述了 Docker 与 DPI 部署有关的底层报文处理技术 DPDK, 并给出了基于 Docker 的 DPI 虚拟化部署方案建议。

关键词: DPI; 网络功能虚拟化; 虚拟网元; Docker

中图分类号: TN929.5

文献标识码: A

doi: 10.11959/j.issn.1000-0801.2019294

Virtualized deployment of DPI based on Docker

PAN Jie, XUE Zhao

China Mobile Group Design Institute Co., Ltd., Beijing 100080, China

Abstract: DPI is a kind of dedicated device for traffic identification and processing in the network of telecom operators, and a large number of DPI devices have been deployed in the network currently. However, the traditional DPI deployment brings the coexistence of multiple DPI systems, duplication of functions, and complicated maintenance and management, and it isn't conducive to the rapid deployment of a large number of standardized DPI applications. With the development of NFV technology, virtualizing DPI deployment is a trend to solve this problem, and Docker is a container-level virtualization technology, DPI can be deployed based on it. Docker and the underlying data packet processing technology DPDK which was related to DPI deployment was described, and the Docker-based DPI virtualization deployment proposals was given.

Key words: DPI, NFV, virtual network element, Docker

1 引言

DPI (deep packet inspection, 深度报文检测) 技术是一种近年在网络中得到广泛运用的技术, 在传统报文检测技术只分析 IP 分组中四元组 (源

IP 地址、目的 IP 地址、源端口号、目的端口号以及协议类型) 的基础上, DPI 还分析应用层信息和识别应用类型。城域网、省网、骨干网、网间互联链路以及国际出口等不同的网络位置都可以部署 DPI 设备, 这种方法满足了网络不同层次的



管理和检测需求,但其需要大量实现相同 DPI 技术的同种专用硬件设备,成本较高,也增加了 DPI 与其他应用系统之间的数据互通难度,因此通过减少采集点和 DPI 设备数量,实现多系统间同一采集点的 DPI 设备共享,力争通过最少的采集点获取最全的数据^[1]是 DPI 技术发展的一个趋势,探索新的 DPI 部署方式也势在必行。

随着网络功能虚拟化 (network function virtualization, NFV) 技术近年来发展的愈发成熟,将 DPI 功能从固化的专用硬件设备中解耦出来,托管在通用标准服务器上作为共享功能,使其向虚拟化演进,是部署 DPI 的一个有效方法。这种方法只需在较少的通用标准服务器上软件化实现 DPI 功能即可,可以大大减少物理设备的数量,有效降低投资成本和能源消耗,也可以使 DPI 和其他应用系统之间的互通与集成更加容易。

2 DPI 的传统部署方式概述

DPI 即对报文中应用层的内容进行深度分析,从而根据其净荷特征识别应用类型或判定合法性。

DPI 设备对网络关键节点处的流量和报文进行检测分析,同时根据事先定义的策略对检测的流量进行过滤控制,能够实现所在链路的业务精细化识别、业务流量的流向分析、业务流量占比统计、业务流量占比整形以及对 DOS 攻击、病毒、木马和滥用 P2P 的过滤控制等功能。DPI 技术的关键在于维护一个高准确性和高实时性的应用特征库,从而保障运营商对应用类型的识别和管控的准确性和实时性^[2]。

在传统的 DPI 部署方式下,DPI 设备在网络中的部署位置随着流量管理和检测要求的不同而不同。常用的 DPI 设备接入方式有串接和并接两种。

- 串接接入方式

直接将 DPI 设备串接到上下游两台核心网设

备之间的链路中,原数据流量全部流经 DPI 设备并完成分析、流控和转发。串接接入方式要求 DPI 设备对原网络实现链路层透明转发,不增加新的路由跳数。

- 并接接入方式

通过在监测链路上部署分光器件将数据流量复制一份到 DPI 设备进行分析,原数据报文不发生改变进行传输。并接接入方式也可以通过交换机端口镜像实现。并接方式通过向业务源端或目的端发送伪造的控制报文,实现在业务流的源端或目的端降低流量或中断该数据流。

目前,运营商广泛采用串接接入方式部署 DPI。然而,传统的 DPI 设备属于专用硬件,功能硬编码在硬件中,耦合程度非常高,不利于新业务的快速开发和部署。同时,由于目前运营商建设的各种应用系统越来越多,对 DPI 的数据检测和控制需求不断增加,导致目前 DPI 系统建设存在多个系统并存的现象,重复性较强,由此带来了投资、维护和管理等多方面的冗余,也给网络的整体安全带来了隐患^[3]。进一步地,随着 DPI 标准的规范,需要部署的是大量标准化的 DPI 应用,同时还要保证快速和良好的隔离性。快速不仅是指配置管理方便,解决环境依赖问题,还在于启动速度^[4]。

3 NFV 网络架构中可嵌入 DPI 虚拟网元的位置

DPI 高度依赖于流量模式和流量动态性,所以使用 NFV 技术实现 DPI 具有更明显的优势^[4]。DPI 领域的知名公司 Sandvine 已经研制出与专有硬件设备性能相当的软件化深度分组检测功能:虚拟深度分组检测功能 (vDPI)^[5]。通过 NFV 技术采用 x86 服务器并用 Docker 虚拟化的方式进行部署,从而实现软/硬件解耦和资源共享,降低设备成本。

NFV 网络的体系架构如图 1 所示,它是一种

丰富的分层架构, 在几个不同的层次都可以嵌入 vDPI。

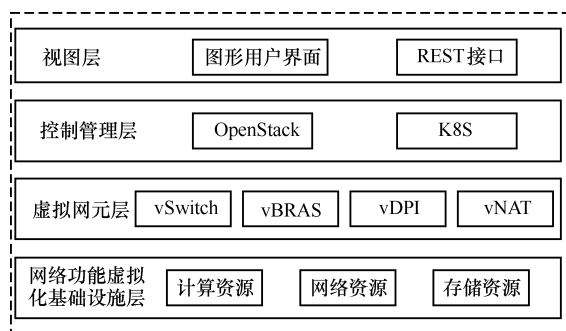


图1 NFV网络体系架构

• 节点层

vDPI 可以放置在网络最前端的节点上运行, 此时, 上层应用接收 vDPI 收集的流量并将控制策略下发给 vDPI。这种嵌入位置的选择可以使 vDPI 最快地接收网络流量并进行分析处理。但是, 由于 vDPI 的数量与网络前端节点保持同步, 所以成本较高。

• 控制层

放置在控制器中的 vDPI 可以连接业务应用层与节点层, 此时, vDPI 成为控制器的重要组成部分, 参与决策的制定, 并为业务应用层提供数据。但是, 网络的可扩展性和分析性能可能会受到影响, 需要运用更高的分布式控制架构来平衡。

• 业务应用层

在业务应用中嵌入 vDPI 相对便捷。此时, vDPI 需要分析处理的是汇聚后的大量数据。由于数据需要经历一定的通信时间才能到达业务应用层中的 vDPI, 因而此时对数据的分析处理会有一定延迟, 因此建立高效的数据分析挖掘模型、提高数据的利用率有很大的必要性。

4 DPI 的虚拟化部署关键技术

DPI 的虚拟化部署涉及的关键技术主要是虚拟化技术的选择以及底层数据分组处理技术。

4.1 虚拟化技术

虚拟化技术是在一台真正的物理机上创建一个或多个虚拟服务器, 虚拟化层要能够直接访问物理资源, 并将被访问的物理资源重新定向到虚拟资源池中。

DPI 的虚拟化部署使用的虚拟化技术主要考虑 KVM 和 Docker。下面对 KVM 和 Docker 进行简要介绍, 并从多个角度对两者进行比较, 总结出 DPI 虚拟化部署使用 Docker 的优势。

4.1.1 KVM、Docker 及其比较

KVM 是一种传统的基于硬件的全虚拟化解决方案, 它完全基于原生 Linux 内核。相对于半虚拟化或准虚拟化, 全虚拟化提供了完整的 x86 通用标准服务器, 包括内存、硬盘空间、CPU、网络适配器等设备和资源。使用 KVM 时, 需要先在 x86 通用标准服务器的自身硬件和 Linux 系统上虚拟出一个运行环境, 并在该运行环境上部署应用, 客户机操作系统和应用程序无需任何修改便可运行于作为服务器的 x86 平台环境中。

Docker 是一种新兴的虚拟化解决方案, 它的基础是 Linux 容器 (LXC), 是基于本机的物理系统实现的。Docker 无需安装更高层的虚拟操作系统, 直接复用本地主机的操作系统, 实现操作系统层面的虚拟化。Docker 虚拟化要素涉及镜像、仓库和容器等。镜像类似于 ISO 镜像文件, 是操作系统的描述模板; 仓库用来存储镜像, 有公有仓库和私有仓库两种形式; 容器则使用镜像创建运行环境。当使用 Docker 时, 只需在当前操作系统上安装所需的容器即可, 各种应用通过引擎部署于容器中, 彼此之间是隔离的。

KVM 和 Docker 的组成构件及组成结构如图 2 所示, 从中可见两者在结构方面的不同。

从特性方面对 KVM 和 Docker 进行了比较见表 1。

4.1.2 Docker 的优势

通过上述分析和比较, 可总结得到 Docker 相

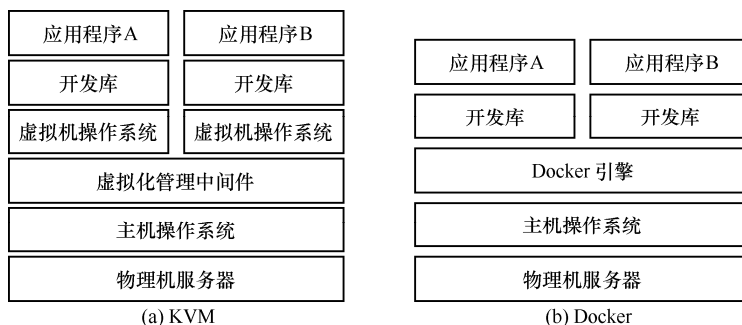


图2 KVM 和 Docker 的组件及结构

表1 KVM 和 Docker 的几个特性比较

特性	Docker	KVM
启动速度	秒级	分钟级
硬盘用量	一般为 MB	一般为 GB
性能	接近原生 Linux	弱于原生 Linux
系统支持量	单机支持上千个容器	一般几十个

比于 KVM 的优势如下。

- 操作速度快

其启动、停止或重启的速度是秒或毫秒级别的，极大地提高了系统的性能。

- 轻量级虚拟化

相对于传统虚拟化一般只能拥有几十个虚拟机而言，使用 Docker 在一台物理服务器上可以轻松创建成百上千个容器，拥有数量庞大的虚拟系统。

- 资源利用率高

Docker 通过镜像复用可以共享操作系统和应用程序组件，极大地减轻系统负载。

相比于 KVM，Docker 作为一种轻量级虚拟化技术，在快速交付部署、实现虚拟化的高效性、迁移扩展以及虚拟网元管理等方面都具有优势，所以基于 Docker 的轻量级虚拟化技术更适合 DPI 部署的虚拟化。

4.1.3 底层报文处理技术

传统 DPI 功能是固化在专用硬件上的，底层网络数据分组的处理也依赖于专有芯片，而 NFV 要求网络设备的功能都基于通用 x86 服务器实现，

其中数据分组的处理主要由网卡和 CPU 完成，但操作系统协议栈复杂的处理流程和虚拟化技术的隔离使数据收发效率较低，因此 DPI 采用虚拟化部署后要实现高性能的数据分组处理需要一个高效的数据分组处理框架，能够适配网卡驱动，绕过操作系统内核协议栈的处理而直接从网卡收发报文。由于 DPI 只关注应用层的载荷数据而无需经协议栈处理得到的状态信息，因此可以绕过内核态与应用态的切换过程，直接将报文交由应用程序处理。

DPDK (data plane development kit, 数据平面开发套件) 是由 6WIND、Intel 等多家公司开发的、用于快速数据分组处理的函数库与驱动集合，主要基于 Linux 系统运行，可以极大地提高数据处理性能和吞吐量，提高数据平面应用程序的工作效率。同时，DPDK 也支持处理 Docker 容器中的报文，提供 Docker 容器环境下的两种报文处理方式，分别是 Slicing 方式和 Aggregation 方式，如图 3、图 4 所示。

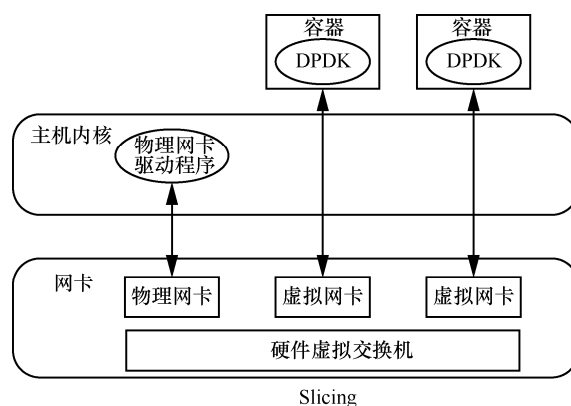


图3 DPDK 处理 Docker 报文的 Slicing 方式

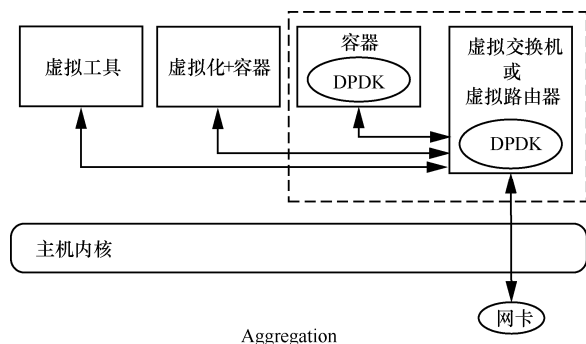


图4 DDPK 处理 Docker 报文的 Aggregation 方式

• Slicing 方式

借助网卡的 SR-IOV (Single Root I/O Virtualization) 功能, 将物理网卡虚拟出多个 VF 虚拟网卡, Docker 容器中的网元通过 DPDK 直接从 VF 虚拟网卡收取报文, SR-IOV 根据网卡混杂模式下的 MAC 地址或者 VLAN 地址将从物理网卡收到的流量调度到对应的 VF 虚拟网卡上。

• Aggregation 方式

由 vSwitch 虚拟交换机或者 vRouter 虚拟路由器对物理网卡的流量进行调度, Docker 容器中的网元和虚拟交换机都运行 DPDK, 虚拟交换机收到物理网卡的报文后根据自身的调度策略将报文调度到 Docker 容器中的网元, 网元通过 DPDK 接收报文。

4.2 具体部署方案建议

采用 Docker 进行 DPI 的虚拟化部署可以考虑以下 3 种方案, 如图 5 所示, 分别描述了 3 种方案各自的部署结构。其核心区别在于主机网卡流

量向不同的 vDPI 调度和分发的策略不同。

• 方案一

Docker 容器中的 DPI 虚拟网元直接接管物理网卡, 通过 DPDK 从物理网卡收取报文。这种方式是一台物理服务器部署一个 Docker 容器, 一个虚拟化容器直接从网卡接受数据分组, 中间使用 DPDK 处理数据面的收发分组, 不需要向各个 Docker 分发, 是一种直通模式。这种方式中的 Docker 能够被编排, 相对于原生 x86 Host 部署, 性能无损耗。

• 方案二

采用标注的 OVS (Open-VSwitch) 从物理网卡收取报文, OVS 负责流量托管和调度, 支持多种人工配置或自动生成策略。这种方式是一台物理服务器部署多个 Docker 容器, 中间使用开源的 OVS 模块向各个虚拟化容器进行数据分发。早期的 OVS 是基于操作系统内核协议栈实现, 在某些应用场景下的数据收发效率低, 因此将 DPDK 集成到 OVS 中进行数据调度分发。该方式是主流部署方法。

• 方案三

因为 DPI 私有定制的数据分发模块 gSwitch 从物理网卡收取报文, 并负责调度到各个 DPI 虚拟网元。这种方式是一台物理服务器部署多个 Docker 容器, 中间使用私有数据分发模块向各个虚拟化容器进行数据分发。该方法基于软件队列

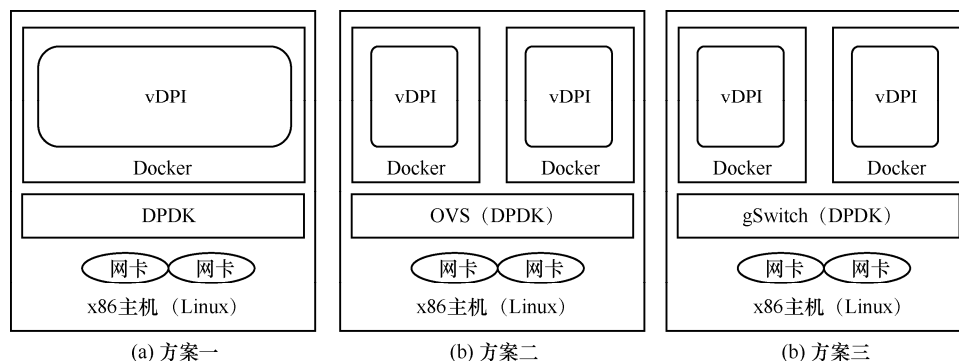


图5 基于 Docker 的 DPI 虚拟化部署 3 种建议方案示意



调度实现流量的调度和分发, DPDK 将报文存在内存报文队列中, gSwitch 从指定的报文队列中取得报文, 这种方式可以减少一次数据拷贝, 性能较高。

其中, 方案二和方案三的区别在于, 方案二使用的是开源数据分发方案, 其依赖于虚拟网卡; 方案三是程序通过 DPDK 从物理网卡收取报文并放入软件队列中, DPI 虚拟网元从软件队列读取数据并进行处理, 其不依赖虚拟网卡; 且方案三中的 gSwitch 模块是为 DPI 私有定制的专属功能, 其比方案二的效率高。

DPI 采用 Docker 进行虚拟化部署后, 可以将 DPI 自身的功能进行分解, 然后分别部署在不同的 Docker 容器中, 在业务层面上实现抽象和解耦。同时, 还可以和其他虚拟网元 (如 vBRAS) 进行灵活的对接和协同工作。随着 DPI 功能的标准化程度越来越高, 借助于简单的脚本工具就可以非常轻松地完成基于 Docker 的虚拟化部署。

5 结束语

随着 NFV 理论的成熟和各种具体虚拟化技术的发展, DPI 的实现和部署也可以逐渐向虚拟化方式演进, 作为一个软件功能部署到 x86 通用标准服务器上, 使其不再依赖于专用硬件, 而通过功能抽象达到软硬件解耦的目的。虚拟化部署的好处可以体现在两方面, 一方面可以大大地减少原有价格昂贵的专用 DPI 硬件设备的数量, 而改用 x86 通用标准服务器后, 其成本低廉, 运营商以往巨大的投资成本能够得到大幅度的降低; 另一方面, 基础设备资源和 API 的开放、共享使快速开发和自动部署新业务成为可能, 且业务需求可以更灵活地弹性伸缩、故障隔离和自愈等。进一步地, Docker 作为新型的虚拟化技术, 其更小的性能损耗、更快的启动速度和更强的可移植性等优越性能使其在 DPI 的虚拟化部署上有更大的优势。

在 DPI 的虚拟化部署中, 还有一个值得研

究的问题是如何更精确地确定 DPI 虚拟网元在网络中的合适位置, 这个问题可能需要考虑软件使用许可费、节能性能和网络带宽等方面的要求和约束。

参考文献:

- [1] 侯慧芳, 潘洁. 大数据背景下运营商建设统一 DPI 系统的思考[J]. 电信科学, 2017, 33(4): 191-197.
HOU H F, PAN J. Discussion of constructing unified DPI system by operators in the background of big data[J]. Telecommunications Science, 2017, 33(4): 191-197.
- [2] 刘超, 刘馨璐, 王攀, 等. 基于 DPI 和大数据分析的宽带家庭画像[J]. 计算机技术与发展, 2018, 28(12): 162-166.
LIU C, LIU X L, WANG P, et al. Broadband family portrait based on DPI and big data analysis[J]. Computer Technology and Development, 2018, 28(12): 162-166.
- [3] 潘洁, 郝卓宁, 林鹏. 基于 SDN/NFV 的 DPI 智能化网络体系的构建[J]. 电信科学, 2015, 31(6): 115-119, 125.
PAN J, XI Z N, LIN P. Construction of DPI intelligent network system based on SDN/NFV[J]. Telecommunications Science, 2015, 31(6): 115-119, 125.
- [4] 伍阳. 基于 Docker 的虚拟化技术研究[J]. 信息技术, 2016(1): 121-123, 128.
WU Y. Research on virtualization technology based on Docker[J]. Information Technology, 2016(1): 121-123, 128.
- [5] BOUET M, LEGUAY J, CONAN V. Cost-based placement of vDPI functions in NFV infrastructures[C]// IEEE Network Softwarization Conference, April 13-17, 2015, London, United Kingdom. Piscataway: IEEE Press, 2015: 490-506.

[作者简介]



潘洁 (1978—), 女, 中国移动通信集团设计院有限公司工程师、高级咨询设计师, 主要研究方向为数据网络。



薛翌 (1992—), 女, 中国移动通信集团设计院有限公司助理工程师、助理咨询设计师, 主要研究方向为数据网络。