面向云的软件定义防御体系研究

胡卫宏 叶崛宇 闫夏莉* 岳巧丽 张海阔
(中国互联网络信息中心 北京 100190)

摘 要 随着云计算的发展,越来越多的传统应用迁移到云上,传统防御体系面对高度灵活的云服务,逐渐左支右绌,疲于应对。围绕云服务安全问题,结合云数据中心的现状,提出多层次细粒度的软件定义防御体系,支撑云服务用户安全弹性定制以及防御策略动态部署。在此体系架构下,进一步提出虚拟化防火墙设计,在数据平面上建立安全插件机制,为云服务用户的流量清洗等逻辑提供高速网络环境。实验结果证明了该体系的技术可行性,并且虚拟化防火墙设计能够满足高性能防御的需求。

关键词 云计算 软件定义防御 虚拟化防火墙 数据平面

中图分类号 TP3 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2020.10.049

CLOUD-ORIENTED SOFTWARE DEFINED DEFENSE SYSTEM

Hu Weihong Ye Jueyu Yan Xiali* Yue Qiaoli Zhang Haikuo
(*China Internet Network Information Center*, *Beijing 100190*, *China*)

Abstract With the development of cloud computing, more and more traditional applications are migrated to the cloud. Traditional defense systems are increasingly difficult to adapt to highly flexible cloud services. Around cloud service security, considering the development status of cloud data center, this paper proposes a multi-level, fine-grained software defined defense system to support flexible security customization as well as dynamic deployment of defense strategies. Under this architecture, the design of the virtualized firewall is further proposed, and a security plug-in mechanism is established in the data plane to provide high-speed network environment for cloud service users' traffic cleaning and other logic. The experimental results demonstrate the software defined defense system is technically feasible and the virtualized firewall design fulfills the needs of high-performance defense.

Keywords Cloud computing Software defined defense Virtualized firewall Data plane

0 引 言

云计算是在数据中心主机上承载多种不同服务,以实现计算、网络、存储等资源的整合与共享,从而达到高效利用资源的目的[1]。为了实现更加高效的数据处理,传统的数据中心应用云计算技术是必然的发展趋势。随着数据中心这种技术架构的转变以及大数据、移动互联网等的快速发展,网络安全的边界概念越来越模糊化,传统意义上的将防火墙部署在网络边界的防御手段已经难以有效地发挥作用。一方面,这种单点防御策略难以在性能上满足数据中心流量与日俱增的需求;另一方面,新的安全威胁与攻击层出不穷,传统防御手段的检测识别与响应处理需要在网络中的多个点(路由器、交换机、防火墙等)进行预先设置,工作量大,配置复杂,无法满足数据中心网络对安全威胁做出快速响应的需求。

软件定义网络[2](Software Defined Network, SDN)是一种新型网络架构,通过将网络设备控制面与数据面分离来实现网络流量的灵活控制,让网络成为一种可灵活调配的资源。基于 SDN 的三个重要概念:可编程、控制平面与数据平面分离、集中式控制模型。文献[3-7]提出了面向 SDN 的软件定义安全防御策略,能够为数据中心网络边界、虚拟层和租户提供灵活的安

收稿日期:2019-06-21。国家自然科学基金项目(61303242)。**胡卫宏**,博士,主研领域:网络通信协议与网络信息安全。**叶崛宇**,硕士。**闫夏莉**,硕士。**岳巧丽**,硕士。**张海阔**,博士。

全保障,但这些方案不能很好地兼容传统架构的数据中心网络。

网络功能虚拟化[8](Network Function Virtualization,NFV)是云计算的一部分,将网络功能从原本的硬件设备上分离出来,以实现弹性、灵活、与设备厂商无关的网络组网。目前,OpenNetVM[9]和 Click-OnOSv[10]等利用 DPDK[11]、NetMap[12]技术已经能够在通用计算架构的虚拟机上实现高性能的网络功能虚拟化。随着互联网的不断发展,对于云服务商而言,各类新型业务层出不穷,采用传统的专用防火墙难以满足新业务在安全防护方面快速开发部署的需求,虚拟化防火墙在云服务中的意义日益显著。本文基于数据与控制平面分离,以及网络功能虚拟化思想,提出一种面向云的软件定义防御体系架构,以解决云数据中心网络传统防御方式的不足,同时提出该体系下一种高性能虚拟防火墙的设计,并通过实验分析证明了该防御体系的可行性。

1 软件定义防御体系

软件定义防御体系(Software Defined Defense System,SDDS)旨在提供一个模块化、灵活、安全的基础性框架,既可支持传统网络安全以及访问控制策略,也能够支持演化到 SDN 网络。SDDS 能够无缝集成到云数据中心网络环境,协同利用内部以及外部的资源,实现可定制化的安全防御策略。

1.1 设计思路

基于云数据中心网络环境的特点,SDDS 的设计应从以下几个出发点考虑:

1)自动化:基于策略驱动,采用软件定义安全方式,由软件来管理,实现安全功能模块(如入侵检测、网络分片和访问控制等)的自动部署与执行。

2)灵活性:功能弹性化,可按需定制,易于规模化应用,能够随着安全威胁或环境变化进行策略调整。对硬件安全设备依赖最小,适用于多种应用环境(如云计算、虚拟架构等),支持对新加入设备自动进行基础安全策略配置与实施。

3)可移植性:将安全策略从硬件层面抽象到软件层面,从而允许安全防御模块在不同环境进行重复部署。

4)协同控制:通过协同管理数据中心网络的安全要素,例如入侵检测、防护、网络分片、防火墙和监控等,实现联动防御。

5)管理可视化:全网拓扑及事件可视,以实现全网络安全态势可感知,便于管理以及对威胁或攻击进行快速响应处理。

1.2 逻辑架构

SDDS 架构的主要特征是通过对网络流量进行数据层的编程实现软件定义的安全防御功能;逻辑上的特征是中心化的控制平面,便于进行灵活的安全管理操作,而不需要对底层网络安全元素个体进行单独配置。此外,全网层面的管理机制保证了不同网络设备之间的高效联动。SDDS 的逻辑架构如图 1 所示。

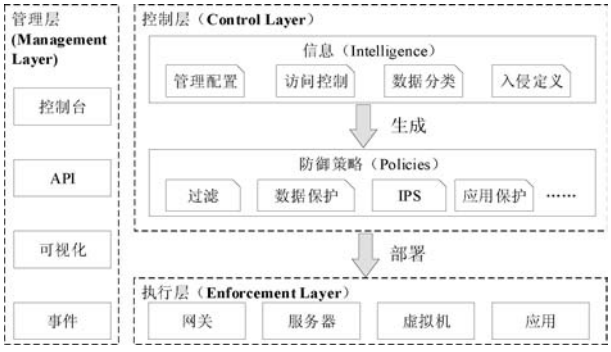

图 1 软件定义防御体系逻辑架构图

执行层(Enforcement Layer)位于数据平面,提供了执行防御保护的一个通用平台。安全策略部署于网络上物理或者虚拟的点,称为执行点,其可以位于网络边界、服务器主机、虚拟机等,能够执行防御动作。通过设置不同的执行点,可对网络进行分片管理,每个分片区域实施不同的安全防御逻辑,以实现灵活高效的防御。

控制层(Control Layer)基于软件定义思想,具有快速适应变化、动态更新策略的特点。控制层是软件定义防御体系的核心部分,负责根据攻击或者威胁事件消息产生软件定义的防御功能,并将其部署到合适的网络位置,包括专用防御设备、运行于主机或者虚拟机的服务和移动设备等。防御的类型包括攻击威胁的防御、访问控制、数据保护等。攻击及威胁的特征信息来自于网络中其他检测设备,访问控制信息来自于管理层配置。

管理层(Management Layer)协同基础架构,为整体框架带来灵活性,是网络管理与其他两个层的接口。其功能包括控制中心、事件处理和可视化等。控制中心可对网络进行分割,以确定防御策略的最佳部署位置。同时负责管理访问控制的定义以及攻击和威胁特征数据的更新,此外还可同意或禁止防御点安全策略的执行。管理的可视化有助于对安全事件的进行主动响应,并感知网络拓扑的改变,从而掌握全网络的安全态势。通过完善威胁检测分析算法,协调数据中心网

络各安全设备的更新策略,实现全网动态化的防御。

SDDS 架构的核心特点是灵活性,主要体现在控制层一方面能够根据新的安全威胁,动态地更新防御策略;另一方面新的检测识别方法或安全技术也能够被及时应用到防御系统之中。执行层仅提供执行防御保护的平台,防御策略由控制层的软件控制,因此底层硬件部署可保持不变。在管理层的全局协调下,SDDS 能够自动适应不断变化的安全威胁或攻击,同时也能够在必要的情况下,例如威胁识别检测可靠性不高,通过网管介入实现定制化防御。

1.3　系统架构

基于 SDDS 的总体设计和逻辑架构,一个典型数据中心网络环境的 SDDS 系统结构如图 2 所示。

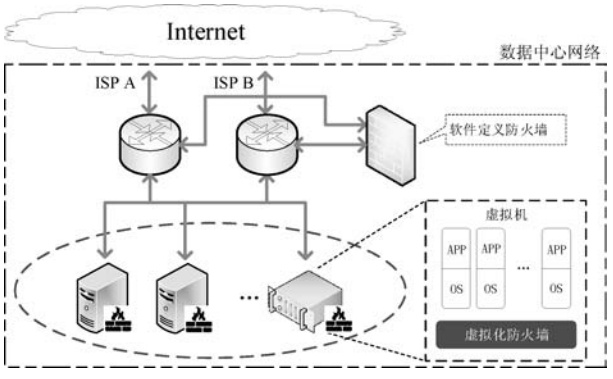

图 2　软件定义防御体系示意图

与传统防御方式类似,数据中心网络边缘部署防火墙集群,通过流量牵引对出入网络的流量按照安全策略进行处理。但在 SDDS 中,该防火墙集群担任全局防御的角色,支持软件定义防御,可对流量进行按需牵引,提高防御性能。同时,它也可作为网络边界网关执行点,部署 SDDS 全局管理模块,协调全网安全资源实现动态按需防御。

此外,在数据中心网络的主机或虚拟机也部署其用于单点防御的虚拟化防火墙。全局软件定义防火墙的部分安全功能被卸载到虚拟化防火墙,以实现高性能、就近、定制化的防御。虚拟化防火墙可在通用架构主机通过虚拟方式实现,除具备灵活性特点外,也可降低部署硬件成本。与逻辑架构对应,虚拟化防火墙包括两个功能实体:控制器与策略执行单元。控制器对分片网络进行管理和控制,能够接收该区域网络中所有的消息,可以给下层执行单元下发安全策略,并对其运行状态进行监控。执行单元是一个实现安全策略动作的实体,运行于数据平面,它对数据流的处理规则由安全策略规定。虚拟化防火墙为主机或者虚拟机租户提供不同粒度的防御,是实现全网层次化、多粒度、灵活部署、动态更新的安全防御体系的重要组成部分。

2　虚拟化防火墙设计

网络功能虚拟化是近年来国内外广泛研究的技术,其核心思想是基于通用软硬件平台实现传统专用设备上的网络功能,以达到资源灵活共享的目的,是实现虚拟化防火墙的基础。基于该技术以及 SDDS 逻辑架构,本文提出一种 SDDS 体系下的高性能虚拟化防火墙设计。

2.1　总体架构

虚拟化防火墙是软件定义防御体系的核心组件,需要满足两方面的基本需求:一是具备防火墙的高性能网络流量处理能力;二是可以隔离不同业务以实现虚拟化。其中具备高性能网络流量处理能力是虚拟化防火墙的首要前提,目前业界比较成熟的高性能网络流量处理框架有 PF_RING[13]、NetMap 和 DPDK 等。Intel 推出的基于 Linux/FreeBSD 的开源开发包 DPDK,以其免费、高性能、良好的硬件支持和社区环境等特点,得到了广泛应用。DPDK 基于数据平面(Data Plane)和控制平面(Control Plane)分离的理念,采用 DDIO(Data Direct I/O)、大页面(Huge Page)、环形缓冲区(Ring Buffer)和 CPU 绑定等技术,实现数据平面的高速网络流量处理。本文基于 DPDK 技术提出一种虚拟化防火墙设计,其数据平面包含网络流量转发器和安全插件,控制平面由操作系统及各种应用软件组成。数据平面和控制平面通过 DPDK 的高速接口 KNI(Kernel NIC Interface)进行通信,虚拟化防火墙总体架构如图 3 所示。

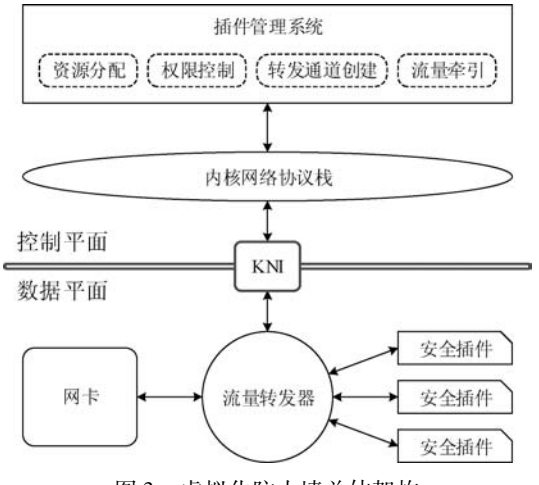

图 3　虚拟化防火墙总体架构

2.2　插件虚拟化

安全插件是不同用户针对各自业务定制的防护软

件,以动态链接库的形式接入虚拟化防火墙,并提供预先定义的标准回调函数接口供流量转发器调用,负责相关网络流量的清洗,对应 SDDS 逻辑架构执行层功能。插件管理系统负责动态加载和卸载安全插件,并为插件提供轻量级的虚拟化机制,在安全插件加载的过程中实现计算、网络和存储资源的虚拟化,兼顾网络性能和隔离安全,对应 SDDS 逻辑架构的控制层。

插件管理系统在加载安全插件的过程中掌控相关资源分配、权限控制、转发通道创建和流量牵引等事务,是虚拟化防火墙的控制中心。插件管理系统加载安全插件的关键流程如下:(1) 为插件 fork 子进程并绑定 CPU,控制子进程不继承父进程的信号、消息队列、定时器等,隔离插件的 CPU、内存空间和 IPC(Inter-Process Communication),实现插件的计算资源虚拟化;(2) 通过 chroot 设置子进程的权限和根路径,隔离插件的用户、组、文件系统,实现插件的存储资源虚拟化;(3) 调用 DPDK 的 EAL(Environment Abstraction Layer)初始化函数接口,指定子进程的类型(--proc-type)为 secondary,通过共享大页面的方式创建与流量转发器之间的高速网络流量转发通道,并为通道配置基于目的 IP 地址的转发规则,隔离插件的网络流量,实现插件的网络资源虚拟化;(4) 循环调用插件的回调函数并根据返回值处理数据报文;(5) 通过 Quagga 进行 BGP 牵引[14],将去往相关目的 IP 地址的网络流量引入虚拟化防火墙。

插件管理系统为安全插件提供虚拟化机制,并将与之相关的网络流量牵引到虚拟化防火墙,流量转发器在此基础上进行网络流量的数据链路层处理与网络层转发。

2.3 流量转发器

流量转发器是数据平面的核心,负责网卡、安全插件和 KNI 之间网络数据流量的高速转发。图 4 为流量转发器的模型示意图。

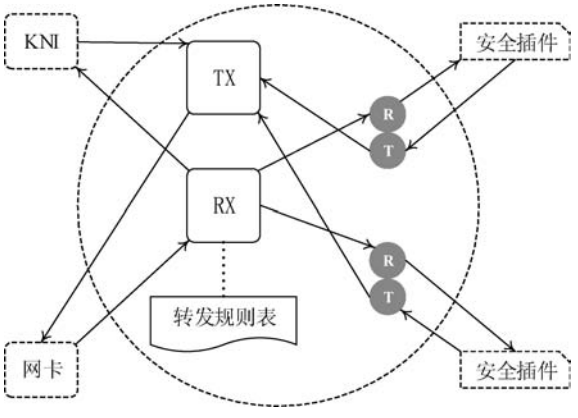

图 4 流量转发器模型示意图

流量转发器通过回调函数将数据报文指针传递给安全插件,并基于函数返回值决定丢弃或转发相应的数据报文。流量转发器绑定两个 CPU(RX 和 TX),并为每个安全插件分配独立的高速网络流量转发通道(R 和 T)。RX 负责从网卡接收网络流量并根据转发规则进行转发,TX 负责将网络流量发送给网卡,R、T 分别为安全插件的输入、输出环形缓冲区。

转发规则表包含插件转发规则和本地转发规则,转发规则由目的 IP 地址和转发目标组成。插件转发规则负责将相关网络流量引入安全插件,目的 IP 地址为其防护的服务 IP,转发目标为其输入环形缓冲区。本地转发规则负责将虚拟防火墙控制平面相关的网络流量(如 BGP 协议)引入本地操作系统,目的 IP 地址为虚拟防火墙的网络接口 IP,转发目标为 KNI。

数据接收 RX 的关键流程如下:(1) 启动相关网口的 checksum-offload 功能,将数据校验工作卸载到网卡以降低 CPU 的负载;(2) 采用 burst 模式从网卡批量接收数据报文,burst 模式可降低每个数据报文的驱动级传输成本(PCIE 总线带宽、函数调用、内存访问等);(3) 根据转发规则确定数据报文的转发目的;(4) 根据转发目的对数据报文进行排序,并采用 burst 模式将数据报文批量压入相关插件的环形缓冲区或 KNI 的 FIFO 队列,burst 模式可以降低每个数据报文的平均 CAS(Compare And Swap)自旋概率及函数调用消耗。与数据接收 RX 类似,数据发送 TX 同样采用 checksum-offload 功能和 burst 模式以提高网络流量的传输性能,其关键流程如下:(1) 从 KNI 批量收集数据报文;(2) 从环形缓冲区批量收集数据报文,并交换报文的源 MAC 和目的 MAC;(3) 设置数据报文的硬件校验标志位,将数据校验操作卸载到网卡;(4) 将数据报文批量发送到网卡。

流量转发器根据转发规则表,完成对网络流量的转发和隔离,并负责数据报文的校验,为安全插件提供网络上下文环境,以进一步实现具体业务的安全防护逻辑。

3 实验验证

随着网络技术的发展,网络攻击越来越多且形式多样,使用防火墙进行流量的访问控制是重要的安全防护之一。网络攻击中的分布式拒绝服务[15](DDoS)是攻击者控制傀儡机器对攻击目标发起大量请求,导致目标服务器资源耗尽无法为用户提供正常服务。DDoS 是云数据中心网络常见的攻击形式,且多为大规模流量攻击,对防御性能要求较高,应用服务软件很难

通过自身的安全策略进行有效防御。本文开发了流量转发器以及插件管理系统等模块,实现了基于 DPDK 的虚拟化防火墙,根据攻击流量特征开发安全插件,利用虚拟化防火墙平台清洗攻击流量以保障正常流量得到及时服务。

测试环境利用一台通用服务器及操作系统搭建,系统配置如表 1 所示。实验以 DNS 协议为例进行测试,使用应用软件 ISC BIND v9.12[16] 搭建权威服务器。按照后缀域名匹配过滤及前缀域名匹配过滤规则开发安全插件,通过插件管理系统进行配置,绑定单个 CPU 接入到虚拟化防火墙中。采用思博伦发包机模拟随机域名攻击场景,对虚拟化防火墙的防御性能进行测试。

表 1　虚拟化防火墙系统配置

配置内容	描述
机型	HPE DL360 Gen9
CPU	Intel Xeon Processor E5-2699 v4
内存	DDR4-2133
网卡	Intel Ethernet Converged Network Adapter X710-DA2
操作系统	CentOS Linux release 7.1.1503 (Core)

搭建应用软件无防护和虚拟化防火墙防护两种环境进行平行测试。测试流量由正常 DNS 查询与攻击流量混合构成。获取.cn 权威服务器正常解析期间的查询日志,提取域名组成正常流量;由子域名随机和后缀域名随机 1∶1混合组成攻击流量。调整正常流量和攻击流量的比例进行发包,并统计记录回应报文,分析权威服务器的服务能力(应用软件的服务能力是指正常查询流量的回应占比),以及虚拟化平台的防御性能。根据本实验环境下应用软件的最大服务能力,固定发送 QPS 为 400 000(正常流量)并逐步增加攻击流量,使总流量最终达到 10 Gbit/s(QPS 约为 1.2×10^7)。通过发包机统计回应报文数据,计算正常流量回应的占比,分析攻击场景下应用软件的服务能力,测试结果如图 5 所示。

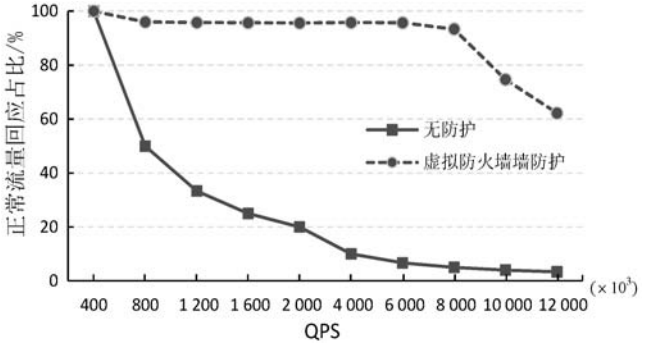

图 5　应用软件的服务能力(单 CPU 插件)

可以看出,应用软件在无防护的场景下服务能力下降明显,当总流量达到 4 000 000 以上时,正常流量回应的占比不足 10%,大部分正常查询都无法得到应答。经过虚拟化防火墙清洗攻击流量后,随着流量逐步加大应用软件的服务能力仅有轻微的下降,但总流量超过 7 000 000 后应用软件的服务能力出现较为显著的下降。经分析,由于小部分正常流量和攻击流量有相同的特征,因此只要安全插件启动,这部分流量就会被误清洗,导致服务能力小幅下降。当总流量超过 7 000 000 后,超过单个 CPU 处理能力的上限导致服务能力下降较为明显,通过将安全插件绑定至多个 CPU 可消除处理能力瓶颈。图 6 为双 CPU 插件的测试结果。

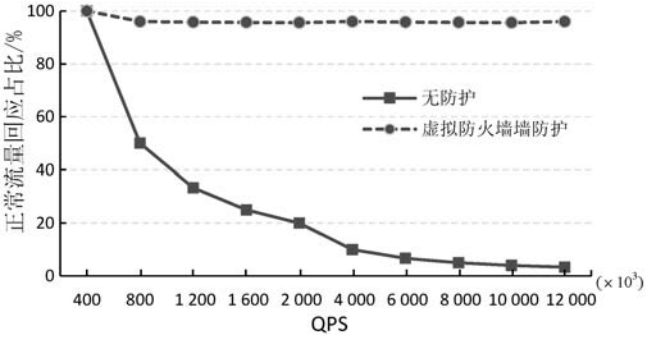

图 6　应用软件的服务能力(双 CPU 插件)

测试结果表明,单个应用服务器的流量处理能力有限,在遇到大规模攻击时服务能力显著下降,虚拟化防火墙在万兆网卡下能处理 10 Gbit/s 的网络流量,最大化保障应用软件的服务能力。虚拟化防火墙方案有较强的可实施性,且配置灵活,适应多元化应用场景,可支撑网络应用服务器根据自身需求定制开发安全插件,利用虚拟化防火墙平台的高性能流量处理能力有效防御网络攻击。

4　分析与讨论

SDDS 灵活的架构设计以及软件定义特性,顺应了云数据中心网络发展趋势,不仅对 DDoS 攻击有良好的防御效果,而且在安全功能、协议支持、业务系统集成中都能够达到传统防御的效果。

4.1　安全功能扩展

访问控制是云数据中心环境下最基础的安全功能需求。外网的访问控制可由部署在网络边界的全局软件定义防火墙实现;云内部多租户的环境中,将同一物理主机中的虚拟机(租户)流量引入虚拟防火墙以实现有效的网络隔离。云环境的动态化特性,对防火墙的入侵检测(IDS)以及防御(IPS)功能也提出很高要求。SDDS 一方面兼容传统的入侵检测系统,全局管理

模块将接收到的入侵事件进行处理并生成特征指纹，再下发到虚拟防火墙执行防御动作；另一方面，虚拟防火墙的软件定义特性允许开发定制化入侵检测算法，并采取过滤、隔离或导流等措施，实现检测、分析与防御的统一，快速对入侵行为做出响应。此外，在更为复杂的 Web 应用防护方面，SDDS 既能够以串联的方式部署在服务器前端，采用传统的规则匹配方式，识别并阻断异常请求，又能够支持基于云的 WAF 防护，采用 DNS 技术，将用户请求转发至云端节点进行检测。

4.2 网络协议支持

在网络协议层面，软件定义防御体系可能遇到的问题主要集中在虚拟化防火墙，具体而言包括两个方面：一是虚拟化防火墙能否对 TCP/IP 模型各层网络攻击进行安全防护；二是虚拟化防火墙能否支持各种常用的管理控制协议，以支撑软件定义防御体系中各个子系统的协同运作。对于第一个问题，DPDK 采用将 PCIE 总线映射到用户空间的机制给予程序直接操作网卡的权限，使得虚拟化防火墙可以存取数据链路层、网络层、传输层和应用层的完整数据，从而实现全协议栈的安全防护。而对于第二个问题，KNI 为数据平面和控制平面之间建立的高速数据转发通道，使得虚拟化防火墙能够与操作系统无缝对接，复用操作系统的所有网络功能，与软件定义防御体系中的其他子系统进行网络交互，实现虚拟化防火墙的可管可控。

4.3 业务系统集成

软件定义防御体系具有高度的灵活性和可扩展性，可在零侵入的情况下集成到各类业务系统中，对已有业务系统影响很小。分析软件定义防御体系的两种部署方式，全局虚拟化防火墙通过旁路部署，旁挂于云数据中心的路由器，利用 BGP 协议进行流量牵引，无须改变已有业务系统的网络拓扑结构；单点虚拟化防火墙取代传统网卡驱动，在完成驱动功能的基础上实现安全防护，对业务系统没有代码侵入，无须改动已有的业务流程。利用网络功能虚拟化技术，虚拟化防火墙中安全策略的应用或升级不影响业务系统，而是采用安全插件的方式动态加载和卸载，方便实时管理和控制，具有实施成本低、技术架构无关性等特点，可灵活应对不同业务系统的安全需求。

5 结 语

本文针对传统防御体系难以满足云服务安全需求的问题，提出软件定义防御体系及虚拟化防火墙模型，为云服务用户提供高度灵活的网络安全保障机制。弹性化是云计算的核心价值，弹性的资源分配和应用部署受到市场的广泛认可。随着云计算的持续发展，相关的网络安全问题日益凸显，探索弹性的新型防御体系，对于完善云的内涵具有积极意义。

参 考 文 献

[1] Armbrust M，Fox A，Griffith R，et al. A view of cloud computing[J]. Communications of the ACM，2013，53(4)：50－58.

[2] Kim H，Feamster N. Improving network management with software defined networking[J]. IEEE Communications Magazine，2013，51(2)：114－119.

[3] 黄润，肖志良. 集成 SDN 框架的启发式数据流调度算法研究[J]. 计算机应用与软件，2019，36(4)：155－160.

[4] 石悦，李相龙，戴方芳. 一种基于属性基加密的增强型软件定义网络安全框架[J]. 信息网络安全，2018(1)：15－22.

[5] Suh M，Park S H，Lee B，et al. Building firewall over the software-defined network controller [C]//16th International Conference on Advanced Communication Technology. IEEE，2014：744－748.

[6] 王鹃，刘世辉，文茹，等. 基于 OpenFlow 的 SDN 状态防火墙[J]. 计算机工程与应用，2018，54(15)：84－90.

[7] 李兆斌，韩禹，魏占祯，等. SDN 中基于机器学习的网络流量分类方法研究[J]. 计算机应用与软件，2019，36(5)：75－79，164.

[8] Han B，Gopalakrishnan V，Ji L，et al. Network function virtualization：Challenges and opportunities for innovations[J]. IEEE Communications Magazine，2015，53(2)：90－97.

[9] Zhang W，Liu G，Zhang W，et al. OpenNetVM：A platform for high performance network service chains[C]//Proceedings of the 2016 Workshop on Hot Topics in Middleboxes and Network Function Virtualization. ACM，2016：26－31.

[10] Marcuzzo L DC，Garcia V F，Cunha V，et al. Click-on-osv：A platform for running click-based middleboxes[C]//2017 IFIP/IEEE Symposium on Integrated Network and Service Management. IEEE，2017：885－886.

[11] Intel. Data plane development kit[EB/OL]. (2014)[2019－06－21]. https://www. dpdk. org.

[12] Rizzo L. Netmap：A novel framework for fast packet I/O[C]//21st USENIX Security Symposium，2012：101－112.

[13] PF_RING documentation[EB/OL]. (2018)[2019－06－17]. https://www. ntop. org/guides/pf_ring.

[14] Jakma P，Lamparter D. Introduction to the quagga routing suite[J]. IEEE Network，2014，28(2)：42－48.

[15] Alieyan K，Kadhum M M，Anbar M，et al. An overview of DDoS attacks based on DNS[C]//2016 International Conference on Information and Communication Technology Convergence (ICTC). IEEE，2016：276－280.

[16] BIND 9[CP/OL]. US：Internet systems consortium.[2019－06－17]. https://www. isc. org/bind/.