

基于SDN的高校计算机网络实验室设计

李月溶, 薛涛, 薛文生

(西安工程大学计算机科学学院, 陕西 西安 710048)

摘要: 网络应用数量的迅猛增长, 让新型网络创新架构SDN受到各界的广泛推崇。为了解决许多高校的计算机网络实验室还不支持SDN相关实验, 对计算机相关专业学生在SDN的学习、应用和研究方面形成阻碍, 不利于学生的系统学习和就业的问题, 本文从SDN的理论建设出发, 基于OpenFlow标准, 搭建基于SDN的计算机网络创新环境实验平台并设计了五种基本网络连接通信情况, 为高校计算机网络实验室提供能够支持SDN相关实验的低成本可行性设计方案。

关键词: 软件定义网络; SDN; 控制器; OpenFlow

中图分类号: TP391.1 **文献标识码:** A

Design and Research of the University Computer Network Laboratory Based on SDN

LI Yuerong, XUE Tao, XUE Wensheng

(School of Computer Science, Xi'an Polytechnic University, Xi'an 710048, China)

Abstract: With the rapid growth of network applications and huge traffic data, network links are congested, and network management is facing severe challenges. SDN has also been noticeable since its birth. However, at present, many computer network laboratories in universities do not support SDN experiments, which hinders the learning, application and research of SDN for computer-related majors, and is not conducive to students' systematic learning and employment. Based on the OpenFlow standard this paper builds an experimental platform of computer network innovation environment based on SDN, and provides a feasible design scheme to support SDN related experiments for computer network laboratories in colleges and universities.

Keywords: software defined network; SDN; controller; OpenFlow

1 引言(Introduction)

随着大数据时代的到来, 网络应用数量的迅猛增长, 用户对流量的需求不断扩大, 封闭的网络设备内置了过多的复杂协议, 增加了运营商定制优化网络的难度, 科研人员无法在真实环境中大规模部署新协议, 一系列问题的激化催生了软件定义网络(Software Defined Network, SDN)^[1]。SDN是一种新兴的网络体系结构, 具有动态性, 可管理性, 自适应性。该架构解耦了网络控制和转发功能, 使网络控制成为可直接编程的, 并为应用程序和网络服务抽象出底层基础架构。SDN可以实现网络虚拟化, 是一种创新网络架构, 最具鲜明的特色在于具备松耦合的控制平面和数据平面、支持集中化的网络状态控制、实现低层网络实施对上层应用的透明, 能够灵活控制网络流量, 使网络能最大程度的接近智能化^[2]。SDN在数据中心、园区、广域网中的持续创新, 使各厂商之间的博弈也将随之进入白热化^[3]。然而目前很多高校的计算机网络实验室只支持传统网络实验, 使学生在学习SDN相关知

识时缺乏实际操作, 这不仅会增加理论知识的学习难度, 也会导致理论知识和实际操作脱节, 遇到实际问题无从下手。另外, 让大学生积极参与SDN技术实践及研究对学生职业规划发展和我国知识人才储备有着积极影响。本文介绍了SDN的相关技术和架构, 详细阐述了如何为计算机网络实验室搭建SDN网络实验环境, 为高校在计算机网络实验室进行SDN相关实验提供思路。

2 SDN发展背景(SDN development background)

随着大数据、云计算及相关技术的发展, 带来了服务器应用需求的爆炸式增长, 众多企业也迎来互联网化、数字化浪潮。虚拟化技术因为能够解决单一物理服务器承载能力有限的问题而更加引人注目。动态地为用户分配所需资源, 使用户能够灵活的进行各种操作是虚拟化技术最重要的特点, 但这样一来也是给网络资源配置增加了负担, 并且虚拟化技术在最大化服务器使用效率的同时也带来了性能损耗^[4]。另一方面, 网络中充斥着很多互相没有联系的网络协议, 这些协

议不能对共性问题进行抽象,这样一来就会为本来繁琐的传统网络增添复杂性,导致网络变得更加脆弱和难以管理。因此,只有尽可能少的网络变动才能勉强维持传统网络的正常运转,无形中给网络管理员进行网络管理增添了巨大压力。

为此,集简化网络配置、优化网络管理、提高网络效率、平滑升级软件等各种优点于一身的新型网络创新架构SDN从诞生之初就备受各界的关注,如今更是受到各界的广泛推崇。SDN使网络行为能够由位于提供物理连接的网络设备之外的软件控制。因此,网络用户可以定制其网络的行为以支持新服务。通过将硬件与软件分离,运营商可以快速地封闭和专有平台的限制中引入创新型和差异化的新服务。从早期的各种开源组织、开源项目、初创公司,到连续四届的开放网络峰会。业界巨头的战略转变,SDN的发展越来越壮大。

现下,每当数据中心网络中基于软件的服务通过因特网协议(IP)地址访问时,就会引发动态配置,而SDN就可以很好的对服务器提供的所有主要资源,如计算容量、内存、存储等进行调控,并且能根据负载性能响应做出相对应的调整,能够提高资源利用率、节省存储空间和加速网络运行。SDN通过创建与物理组件执行相同角色的虚拟设备来解决网络可编程性问题。当给出网络命令的信道可以与用户数据交换的信道分开时,编程网络进行自适应调整变得非常容易,使得诸如超收敛的方法成为可能。

3 SDN网络体系架构(SDN network architecture)

在SDN网络体系架构中,网络的控制平面和数据平面相互分离,并通过南向协议进行通信,使得逻辑集中的控制器可以对分布式的数据平面进行编程控制^[5]。SDN颠覆了OSI的七层架构,将网络划分为转发层、控制层和协同应用层,在SDN技术上运行的服务和应用程序是从提供网络控制的物理连接的底层技术和硬件中抽象出来的。应用程序将通过API与网络交互,而不是与硬件紧密耦合的管理接口。如图1所示。

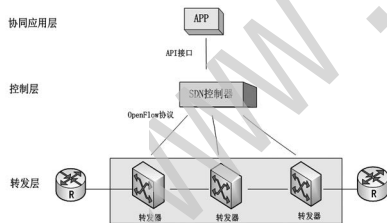


图1 SDN网络架构图

Fig.1 SDN network architecture diagram

协同应用层是实现用户目的的各种层应用程序集合,统称为协同层应用程序。对于开发者而言,这一层是开放区域,供开发者进行开发各种创新应用。也为企业和数据中心网络提供各种端到端的解决方案。需要特别说明的是,转发平面、控制平面和管理平面并不是SDN独有的。相比于同样拥有这三个特质的传统网络,SDN能够对它们进行集中控制,而这一点是传统网络无法做到的。控制层是SDN控制器管理网络的基础设施,主要负责网络路径交换的管理和边界业务路由的生成,并负责维护不同的网络信息、拓扑细节。控制器包含大量业务逻辑,维护网络的全局视图,应用程序

和策略引擎将其视为单个逻辑交换机。转发层也叫基础设施层,主要由转发器和连接器的线路构成基础转发网络,这一层负责执行用户数据的转发,转发过程中所需要的转发表项是由控制层生成的。

在SDN架构中,SDN剥离了网络的控制平面与数据平面,通过增加对网络的可编程性来革新当前偏重静态、配置复杂、改动麻烦的网络架构。SDN允许网络管理员通过动态的自动化SDN程序快速配置、管理、保护和优化网络资源,他们可以自行编写代码,因为程序不依赖于专有软件。数据平面也不必具体实现各种网络协议,只需接收控制平面的操作指令并执行即可实现控制功能。SDN还有一个非常大的优点是它通过开放标准实施时简化了网络设计和操作,控制器提供的指令代替了多个供应商特定的设备和协议。它不属于某一家商业公司,而是属于所有IT企业和一些标准组织,因此SDN的发展也可以打破目前一些网络巨头的垄断并为网络技术的飞速发展提供动力。

4 SDN南向接口和OpenFlow(SDN south facing interface and OpenFlow)

在SDN架构中,网络的控制平面和数据平面相互分离,并通过南向协议进行通信,使逻辑集中的控制器能够可以对分布式的数据平面进行编程控制^[6]。南向接口是为网络数据平面提供统一开放和具有更多编程能力的接口,控制器可以利用这些接口对数据平面设备进行编程控制和指导网络流量的转发。目前OpenFlow是标准化组织ONF唯一确定的控制器南向接口,在SDN发展中举足轻重^[7]。

南向接口对于SDN架构的演进非常重要。在一个真实网络应用方案中,数据平面的南向接口能力决定了SDN方案用户编程能力的上限。在南向接口中,应用最多的是OpenFlow协议,大多数SDN设备和控制器都对OpenFlow支持。相比传统网络的配置协议,支持OpenFlow的网络设备给网络管理员提供了一组更细粒度的可编程API。网络运维人员通过OpenFlow协议可以自定义网络设备的转发行为。但是OpenFlow交换机只是定义了一种通用转发抽象模型。最理想的SDN数据平面是协议无关的通用可编程网络处理模型。

传统的网络体系结构已经不能适应企业、运营商和网络终端用户的需求。创新网络架构SDN已经成为新潮流。在高校计算机网络实验室中利用支持OpenFlow协议的OVS搭建SDN实验环境无疑是最佳选择。

5 SDN计算机网络实验室设计(Design of the SDN computer network laboratory)

搭建支持SDN实验的计算机网络实验室有助于学生在SDN实际应用场景中的学习,对SDN的理论学习具有指导意义。

5.1 利用Mininet搭建OpenFlow环境

Mininet是由一些虚拟的终端节点、交换机、路由器连接而成的一个网络仿真器,它采用轻量级的虚拟化技术让系统可以尽可能的接近真实网络。采用Mininet平台在计算机网络实验室来搭建OpenFlow环境最重要的原因是,在Mininet中得到验证的代码程序可以无缝迁移到真实的硬件环境中,因此非常适合教学研究。目前Mininet官方已经对各个版本

的OpenFlow协议进行了测试,并把Mininet集成到了支持OpenFlow协议的开源控制器环境中。本文设计平台中包括:VMware Player上的Mininet虚拟机、作为客户端的putty、Xming(X-Server),其中putty能够以图形化输出的方式运行程序,便于观察到图形化输出结果。搭建步骤如下:

(1)启动虚拟机程序中的Mininet虚拟机镜像并登录到Mininet虚拟机中。

(2)找到虚拟机的IP地址,建立与Mininet虚拟机的SSH会话,启动Wireshark作为后台进程。

(3)在Wireshark中选择所用的数据包捕获设备,或者选择环回的网络接口,捕获和OpenFlow相关的流量。

(4)建立好Mininet仿真环境后,用OpenFlow控制器和交换机启动OpenFlow协议的通信,如果能在Wireshark窗口查看到捕获的数据包(图2),则表示利用Mininet搭建OpenFlow环境成功。

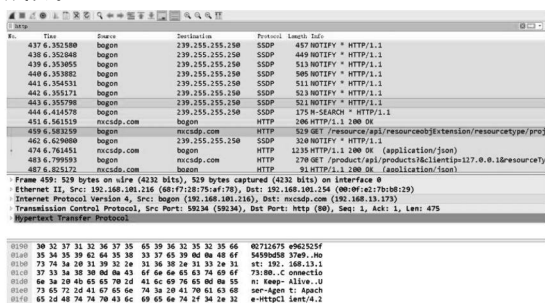


图2 Wireshark展现捕获数据包图

Fig.2 Captured packet map in the Wireshark window

5.2 OVS构建虚拟网络

OVS(OpenvSwitch)是一个虚拟交换软件,开发人员通过代码编辑能够控制网络连接,实现网络自动化。传统的路由器基本需要处理所有的网络路由功能,使用的路由协议也是在硬件中实现,网络管理人员只能通过改变协议的参数或者主动选择哪些协议用于路由。和传统的路由器相比,OVS的结构更加简单,它只考虑包的转发,而没有考虑任何路由规则或者转发规则,所有的转发规则由FlowTable定义,而路由功能由外部的controller中部署的应用程序处理并更新FlowTable。OVS模块管理架构如图3所示。

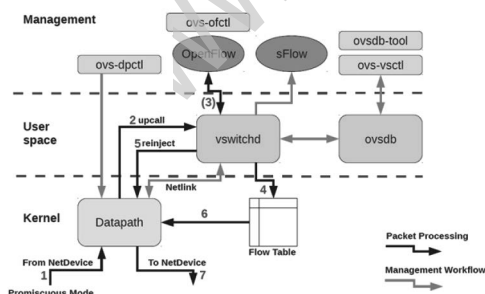


图3 OVS模块管理架构图

Fig.3 OVS module management architecture diagram

从图3可以看出,OVS可以划分为三大块:数据面、控制面和管理面。数据面就是以用户态的OVS-vswitchd和内核态的datapath为主的转发模块,以及与之相关联的数据库模块OVSdb-server,控制面主要是由OVS-ofctl模块负责,基于

OpenFlow协议与数据面进行交互。而管理面则是由OVS提供的各种工具来负责,这些工具的提供也是为了方便用户对底层各个模块的控制管理,提高用户体验。

高校计算机网络实验室应不仅能支持简单网络实验,还应支持各种复杂网络的实验以供学生学习研究。由于复杂网络多种多样,因此本文抽象出最基本的五种网络情况,通过这五种模式的组合就可以构建出不同复杂度的网络,为各种SDN实验提供支持。

(1)构建物理机和物理机相互连接的网络,网络结构如图4所示。

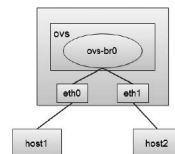


图4 两个物理机互相连接网络图

Fig.4 Two PMs interconnection network diagram

OVS-br0表示OVS的网桥,eth0和eth1表示主机上的网卡,并且这个主机安装了OVS,host1、host2则是两台物理机。两个网卡一端和OVS的网桥连接,一端和物理机连接,使物理机之间可以互通。

(2)构建虚拟机与虚拟机相连的网络,网络结构如图5所示。

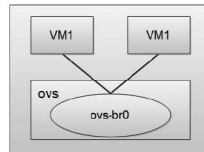


图5 两个虚拟机互相连接网络图

Fig.5 Two VMs interconnection network diagram

OVS-br0表示OVS的网桥,VM1和VM2是两台虚拟机。两个虚拟机只要都和OVS网桥相连就可以实现互通。

(3)构建虚拟机与物理机相连的网络,网络结构如图6所示。

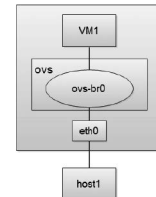


图6 虚拟机和物理机互相连接网络图

Fig.6 VM and PM interconnection network diagram

虚拟机和物理机之间通过OVS网桥和网卡可实现互通。

(4)构建网桥和网桥相连的网络,网络结构如图7所示。

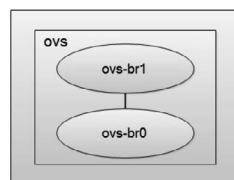


图7 网桥和网桥互相连接网络图

Fig.7 Network bridge interconnection network diagram