

软件定义网络(SDN)研究进展

摘要: SDN 将数据平面与控制平面解耦合, 简化了网络管理。

中文引用格式: 张朝昆, 崔勇, 唐嵩祯, 吴建平. 软件定义网络(SDN)研究进展. 软件学报, 2015, 26(1): 62-81. <http://www.jos.org.cn/1000-9825/4701.htm>

英文引用格式: Zhang CK, Cui Y, Tang HY, Wu JP. State-of-the-Art survey on software-defined networking (SDN). Ruan JianXue Bao/Journal of Software, 2015, 26(1): 62-81 (in Chinese). <http://www.jos.org.cn/1000-9825/4701.htm>

随着网络规模的不断扩大, 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规模部署新协议。

2009 年, McKeown 教授正式提出了 SDN 概念利用分层的思想, SDN 将 **数据** 与 **控制** 相分离; 在控制层, 包括具有逻辑中心化和可编程的控制器, 可掌握全局网络信息, 方便运营商和科研人员管理配置网络和部署新协议等; 在数据层, 包括哑的(dumb)交换机(与传统的二层交换机不同, 专指用于转发数据的设备)。交换机仅提供简单的数据转发功能, 可以快速处理匹配的数据包, 适应流量日益增长的需求。

两层之间采用开放的统一接口(如 OpenFlow 等)进行交互。控制器通过标准接口向交换机下发统一标准规则, 交换机仅需按照这些规则执行相应的动作即可。

1. SDN 体系结构

由于 SDN 标准接口机制确保层次之间既保持相对独立, 又能正常通信, 因此, 标准接口设计的好坏是 SDN 成功设计的关键。

1.1 SDN 诞生发展的背景

4D 架构将可编程的决策平面(即控制层)从数据平面分离, 使控制平面逻辑中心化与自动化, 其设计思想产生 SDN 控制器的雏形。

借鉴计算机系统的抽象结构, 未来的网络结构将存在转发抽象、分布状态抽象和配置抽象这 3 类虚拟化概念:

- 转发抽象剥离了传统交换机的控制功能, 将控制功能交由控制层来完成, 并在数据层和控制层之间提供了标准接口, 确保交换机完成识别转发数据的任务;
- 控制层需要将设备的分布状态抽象成全网视图, 以便众多应用能够通过全网信息进行网络的统一配置;
- 配置抽象进一步简化了网络模型, 用户仅需通过控制层提供的应用接口对网络进行简单配置, 就可自动完成沿路径转发设备的统一部署;

网络抽象思想解耦了路径依赖, 成为数据控制分离且接口统一架构(即 SDN)产生的决定因素。

1.2 体系结构概述

SDN 由下到上(或称由南向北)分为数据平面(也叫做基础设施平面)、控制平面和应用平面。数据平面与控制平面之间利用 SDN 控制数据平面接口(control-data-plane interface, 简称 CDPI)进行通信, CDPI 具有统一的通信标准, 目前主要采用 OpenFlow 协议。控制平面与应用平面之间由 SDN 北向接口(northbound interface, 简称 NBI)负责通信, NBI 允许用户按实际需求定制开发。

- 数据平面由交换机等网络元素组成, 各网络元素之间由不同规则形成的 SDN 网络数据通路形成连接。
- 控制平面包含逻辑中心的控制器, 负责运行控制逻辑策略, 维护着全网视图。控制器将全网视图抽象成网络服务, 通过访问 CDPI 代理来调用相应的网络数据通路, 并为运营商、科研人员及第三方等提供易

用的NBI,方便这些人员订制私有化应用,实现对网络的逻辑管理。

- 应用平面包含着各类基于 SDN 的网络应用,用户无需关心底层设备的技术细节,仅通过简单的编程就能实现新应用的快速部署。

CDPI 负责将转发规则从网络操作系统发送到网络设备,它要求能够匹配不同厂商和型号的设备,而并不影响控制层及以上的逻辑;NBI 允许第三方开发个人网络管理软件和应用,为管理人员提供更多的选择。

网络抽象特性允许用户可以根据需求选择不同的网络操作系统,而并不影响物理设备的正常运行。

1.3 开放式接口与协议设计

SDN 中的接口具有开放性,以控制器为逻辑中心,南向负责与数据层通信,北向负责与应用层通信。由于单一控制机制容易造成控制节点失效,严重影响性能,可采用多控制器方式。多控制器之间将采用东西向通信方式。开放式接口的研究,必然进一步推动 SDN 的深入发展。

由于控制层与数据层解耦合,使得针对这两层的改进相对独立,在层与层之间仅需提供标准南向接口即可。逻辑上,它既要保证数据层与控制层之间的正常通信,又要支持两层独立更新;物理上,设备生产厂商需要开发支持这种标准接口的设备,因为传统网络设备是不能在SDN 网络之中运行。因此,研发南向标准接口成为SDN 基础研究中的重要内容之一。许多组织着手制订南向标准接口。ONF 提出的 CDPI 成为了主流南向接口,它采用 OpenFlow 协议。

除了南向接口相关研究之外,控制器北向接口及控制器间东西向接口同样是研究重点。北向接口负责控制层与各种业务应用之间的通信,应用层各项业务通过编程方式调用所需网络抽象资源,掌握全网信息,方便用户对网络配置和应用部署等业务的快速推进。然而,由于应用业务具有多样性,使得北向接口亦呈现多样性,开发难度较大。然而,这些标准仅对功能作了描述,而未详细说明实现方式。因此,如何实现统一的北向接口标准,成为业界下一步主要推动的工作。

与南北向接口通信的方式不同,东西向接口负责控制器间的通信。由于单一控制器性能有限,无法满足大规模SDN 网络部署,东西向接口标准的制订使控制器具有可扩展能力,并为负载均衡和性能提升等方面提供了技术保障。

2. 数据层关键技术研究

在SDN 中,数据层与控制层分离,交换机将繁重的控制策略部分交由控制器来负责,而它仅根据控制器下发的规则对数据包进行快速转发。为了避免交换机与控制器频繁交互,双方约定的规则是基于流的,而并非基于每个数据包的。

相关技术研究主要集中在交换机和转发规则方面:

- 首先是 交换机设计研究,即设计可扩展的快速转发设备,它既可以灵活匹配规则,又能快速转发数据流;
- 其次是 转发规则的相关研究,例如规则失效后的一致性更新问题等。下面详细讨论数据层相关的研究成果;

2.1 交换机设计问题

SDN 交换机位于数据层面,用来负责数据流的转发。通常可采用 硬件 和 软件 两种方式进行转发。对于硬件来说,具有速度快、成本低和功耗小等优点;利用硬件处理数据,可以保证转发效率,但亟需解决处理规则不够灵活的问题。

为了使硬件能够灵活解决数据层的转发规则匹配严格和动作集元素数量太少等限制性问题,Bosshart 等人针对数据平面转发提出了 RMT 模型。作为另一种利用硬件灵活处理技术,FlowAdapter 采用交换机分层的方式来实现高效、灵活的多表流水线业务。

与利用硬件设计交换机的观点不同,虽然软件处理的速度低于硬件,但是软件方式可以最大限度地提升规则处理的灵活性,同时又能避免由于硬件自身内存较小、流表大小受限、无法有效处理突发流等问题,因而同样受到学术界的关注。

此外,在数据平面中,哪些元素可以交给硬件处理,哪些元素可以交给软件处理,也是值得考虑的问题。

2.2 转发规则相关研究

与传统网络类似,SDN 中也会出现网络节点失效的问题,导致网络中的转发规则被迫改变,严重影响了网络的可靠性。

将配置细节进行抽象,使管理人员能够使用较高抽象层次(high-level)的管理方式统一更新,就可防止低层管理引起的不一致性问题。一般采用两阶段提交方式来更新规则:

- 第 1 阶段,当某个规则需要更新时,控制器询问每个交换机是否处理完对应旧规则的流,并对处理完毕的所有交换机进行规则更新;
- 第 2 阶段,当所有交换机都更新完毕时,才完成更新,否则将取消该更新操作。

3. 控制层关键技术研究

控制器是控制层的核心组件,通过控制器,用户可以逻辑上集中控制交换机,实现数据的快速转发,便捷安全地管理网络,提升网络的整体性能。

本节首先详细阐述了以 NOX 控制器为基础的技术改进方法:一种是采用多线程的控制模式,另一种是通过增加分布式控制器数量,实现扁平式和层次式控制模式。

3.1 控制器设计问题

控制器的基本功能是为科研人员提供可用的编程平台。最早且广泛使用的控制器平台是 NOX,能够提供一系列基本接口。用户通过 NOX 可以对全局网络信息进行获取、控制与管理,并利用这些接口编写定制的网络应用。

网络中可采用两种方式扩展单一集中式控制器:一种是通过提高自身控制器处理能力的方式,另一种是采用多控制器的方式来提升整体控制器的处理能力。

控制器拥有全网信息,能够处理全网海量数据,因此需要具有较高的处理能力。NOX-MT 提升了 NOX 的性能,它是具有多线程处理功能的 NOX 控制器。NOX-MT 并未改变 NOX 控制器的基本结构,而是利用传统的并行技术提升性能,使 NOX 用户可以快速更新至 NOX-MT,且不会由于控制器平台的更替产生不一致性问题(单机多线程处理数据)。

但是对于大规模网络来说,仅依靠多线程处理方式将无法保证性能。通过扩展控制器的数量可以解决上述问题,也就是将控制器物理分布在网络中,仅需保持逻辑中心控制特性即可。这样可使每个交换机都与较近的控制器进行交互,从而提升网络的整体性能。

分布式控制器一般可采用两类方式进行扩展,分别是扁平控制方式和层次控制方式:

- 对于扁平控制方式,所有控制器被放置在不相交的区域里,分别管理各自的网络。各控制器间的地位相等,并通过东西向接口进行通信。
- 对于层次控制方式,控制器之间具有垂直管理的功能。也就是说,局部控制器负责各自的网络,全局控制器负责局部控制器,控制器之间的交互可通过全局控制器来完成。

3.2 接口语言

科研人员致力于开发一种抽象的、高级配置语言。这些抽象语言能够统一北向接口,改善接口的性能,从而全面降低网络的配置成本。

- 耶鲁大学团队开发了一系列网络配置语言,旨在搭建具有优化性能的通用北向接口。Nettle 是描述性语言,采用了函数响应式编程(functional reactive programming,简称 FRP)方式。
- 康奈尔大学与普林斯顿大学联合研究团队首先提出了 Frenetic,与 Nettle 等语言类似,它也具有描述性和 FRP 特性。

3.3 控制层特性研究

控制层存在一致性、可用性和容错性等特性,而所提到的这3种特性的需求无法同时满足,达到三者之间的平衡,是今后的重要工作之一。

1. 一致性:集中控制是SDN区别于其他网络架构的核心优势之一,通过集中控制,用户可以获取全局网络视图,并根据全网信息对网络进行统一设计与部署,理论上保证了网络配置的一致性问题.然而,分布式控制器仍然具有潜在的不一致性问题。
2. 可用性:规则备份可以提升网络的可用性。控制器作为SDN的核心处理节点,需要处理来自交换机的大量请求,而过重的负载会影响SDN的可用性.利用分布式控制器可以平衡负载,提升SDN的整体性能。
3. 容错性:与传统的互联网类似,SDN同样面临着网络节点或链路失效的问题.然而,SDN控制器可以通过全网信息快速恢复失效节点,具有较强的容错能力。

4. SDN 应用研究

- 企业网与校园网;
- 数据中心与云;
- 广域网;
- 无线网络;

5. 未来工作

1. SDN 可扩展性研究;
2. SDN 规模部署与跨域通信;
3. 传统网络与 SDN 共存问题研究;
4. SDN 在数据中心的应用研究;
5. 借鉴 SDN 思想融合IPv6 过渡机制;
6. SDN 与其他新型网络架构融合;
7. SDN 网络安全;