一种SDN网络路径异常监控方法

李 阳 蔡志平 夏 竟
(国防科学技术大学计算机学院 长沙 410073)

摘 要 在SDN网络中,及时掌握网络中数据传输路径的运行状态对于控制器进行网络安全监控和流量的负载均衡至关重要。利用SDN网络架构的特点,提出了一种主动测量与被动测量相结合的路径异常监控方法,以对网络路径的运行状态进行异常监控。从路径的延迟和可用带宽两个方面分析网络路径的运行情况,并对出现异常的路径进行检测和报警,保障网络的正常运行。实验结果表明,在不同的网络条件下,所提方法都能有效测量网络路径的延迟和可用带宽,并且能够及时发现网络中出现的路径异常情况。

关键词 SDN,延迟测量,可用带宽测量,路径异常监控
中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2017.07.005

Method of Software Defined Network Path Abnormity Monitoring

LI Yang CAI Zhi-ping XIA Jing
(College of Computer, National University of Defense Technology, Changsha 410073, China)

Abstract In the SDN architecture, the controller has to master the operating state of network data transmission path segment, which is quite significant for controller to implement the network security monitoring and traffic load balancing. Based on the characteristics of SDN architecture, the paper proposed an approach which combines reactive measurement and passive measurement for monitoring the network path segment abnormity. The method analyzes the operating state of network traversing path from two aspects of path's latency and available bandwidth, which can also perform the detection and give an alarm for the abnormal network path segments. Our experiments show that the method can not only measure the latency and available bandwidth of path segment, but also can detect the abnormal path segment out timely and send out alarm signals. In network conditions of constant and variable network cross traffic, the approach can monitor the abnormal path segment and measure the path's performance parameters efficiently.

Keywords SDN, Latency measurement, Available bandwidth measurement, Path abnormity monitoring

在传统的IP网络当中,控制面与数据转发面具有很高的耦合性,每一个子网络都有自己的控制设备,普通的用户想获取整个网络的路径状态信息几乎是不可能的,而且出于安全的考虑,传统的网络中一些设备不允许用户在其中随意放置数据采集监测点。所以一些基于数据包采集的被动的测量方式很难在传统网络中得到应用。另外,复杂的异构网络中的不同需求使得传统网络的构建和管理过程非常繁琐,传统的IP网络已经达到了一种僵化的程度,很难再向其中增加新的功能。

最近,软件定义网络(Software Defined Network,SDN)[1]已经引起了许多硬件厂商和科研机构的高度重视,因为它可以通过接口的方式实现更加细粒度的网络监控管理和控制。一些优化的网络算法也可以在SDN网络中得到应用,SDN网络架构的设计思想是将传统的网络层次抽象成控制面和数据转发面,控制面的主要工作由控制器完成,其对应的功能主要是进行网络的拓扑管理和对网络中的数据流的转发进行控制。数据转发面的主要工作由交换机来完成,其功能是根据本地流表对数据报文进行匹配转发处理。

网络路径异常监控对于控制器实时掌握网络传输路径的状态意义重大,网络管理者可以根据控制器提供的网络路径的状态信息,对数据流的传输进行路径的优化与调整,合理地进行网络资源的分配。开发者也可以根据网络的路径信息为应用程序设计出合理且高效的传输方式。对网络路径进行测量和监控,可以帮助科研人员掌握网络的运行状态和资源消耗情况,发现网络的瓶颈和潜在的危险,并通过优化网络结构来改善网络的性能,增强网络的健壮性和安全性。

本文在充分利用SDN网络特点的基础上,通过控制器的全局网络拓扑功能和向第三方应用程序提供的编程接口以及OpenFlow协议[2]原有的消息类型,开发和部署了网络路径测量及监控模块,采用主动测量和被动数据采样的方式实现了路径的延迟和可用带宽的性能指标的测量和分析,从而为SDN网络提供了一种可软件定义的测量和监控方案。

到稿日期:2016-08-20 返修日期:2016-10-25 本文受国家自然科学基金项目(61379145)资助。
李 阳(1990-),男,硕士,主要研究方向为未来网络、网络测量,E-mail:liyang06200910@163.com;蔡志平(1975-),副教授,主要研究方向为未来网络、网络安全等,E-mail:zpcai@nudt.edu.cn(通信作者);夏 竟(1980-),男,博士,主要研究方向为计算机网络、网络安全等。

1 相关工作

网络测量的方式一般可以分成被动测量方式[8]和主动测量方式[7]。被动测量方式主要通过网络数据采样的方法来获取网络路径的性能指标，不需要向网络发送额外的探测数据包，基本上不会产生多余的网络负载，也就不会影响网络的性能；但是被动测量需要在网络设备中安装监测点来统计网络状态信息，这并不适合于所有的网络设备，而且有些方法需要硬件的支持，会提高测量成本。主动测量的方式通过向网络中注入额外的探测数据包来监控网络路径的性能指标，例如：流行的网络测量方式 ping 命令就是向网络中注入 ICMP 数据包来进行端到端的网络路径状态的测量。主动测量和被动测量[11]两种方式都可以用来进行网络路径的测量和监控以及收集网络中的统计信息。然而，不同的网络环境需采用适当的测量方式，一方面，主动测量比较灵活，可以根据需要随时发送探测数据包，完成测量和监控过程。但是，主动测量会引入额外的网络负载，对测量结果的准确性有影响；并且由于探测数据包在网络中的行为不可控性会造成无效的测量数据，从而降低测量精度。另一方面，被动测量不够灵活，测量样本难以预知，可能会因为不能收集到足够的测量样本而无法完成测量过程。

在 SDN 的网络测量方面，国内外的研究者都提出了自己的解决方案。其中，一种被动的测量方式 FlowSense[3] 只使用 OpenFlow 原有消息来对整个网络的带宽使用率进行监控，即使这种监控方式不会给网络带来任何的负载流量，但是在动态的网络环境当中，其测量和监控的准确率明显降低，不适用于真实的网络。另外，PayLess[4] 和 MonSamp[5] 提出了自适应的采样算法来适应当前的网络负载，但是它们的采样方式存在冲突，因为 PayLess 方案认为在高流量负载的网络中应该提高数据采样率来保证测量和监控的准确性，然而，MonSamp 则建议在高流量负载的网络中应当降低数据采样率，在更高的流量负载网络中应该降低数据采样给网络带来的负载。程光等[13]提出了一种单链接的链路故障诊断方法，通过 OpenFlow 消息对网络节点的信息进行采集，从拓扑管理、链路拥塞等方面进行链路故障的诊断虽然在一定程度上达到了对单条链路的故障进行诊断的目的，但是此方法过于依赖样本数据采集量，在很大程度上增加了南向接口传输的资源消耗，不适用于大规模的网络环境。

文献[6]提出了一种丢包率与延迟测量的方法，在丢包率的测量过程中，此方法通过不断轮询进出口交换机的流表计数器来获取两个数据值之间的差异，从而实现基于数据流的丢包率的测量工作。在进行链路延迟的测量工作中，它利用 SDN 控制器沿着被测量的路径发送探测数据包，然后这些探测数据包循环地回到控制器。这种方法能够利用往返时间延迟计算出给定路径的延迟。

从上述文献来看，大量的工作主要集中在网络链路性能参数的获取方面，但是关于 SDN 中网络路径的异常监控方法的设计相对较少，主要的测量方法大多都只利用了一种方式，或者是主动测量，或者是被动测量。本文所提方法结合主动测量和被动数据采样，利用 SDN 网络架构的特点和 OpenFlow 协议原有的消息类型，对 SDN 网络中的路径异常进行监控。

2 路径异常监控方法的设计

控制器是 SDN 网络架构的核心组成部分，负责网络中的拓扑发现和管理、网络数据流转发时的路由选择、流量负载均衡以及防火墙方面的工作。控制器拥有全局网络视图功能，掌握着整个网络中网络元件(交换机、链路、主机等)的基本信息。利用控制器的上述特点，在控制器上开发和实现路径异常监控方法对应的功能，其模块结构如图 1 所示，该方法主要包含 4 个功能模块：网络拓扑管理模块、路径延迟测量模块、路径可用带宽测量模块和路径异常监控模块。

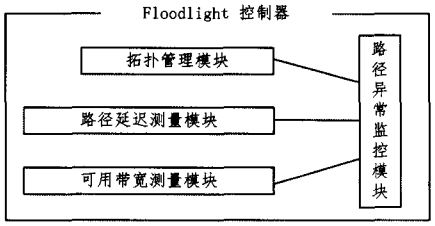

图 1 路径异常监控方法的模块组成

2.1 网络拓扑管理模块

网络拓扑结构管理模块是通过 Floodlight 控制器[9]向第三方应用提供的编程接口(REST API)获取的控制器中保存的网络拓扑结构。在 Floodlight 控制器中，主要有两个模块负责网络结构的拓扑管理工作，分别是链路发现管理模块(Link Discovery Manager)和拓扑管理模块(Topology Manager)。其中链路发现管理模块负责网络结构中交换机之间的链路发现；拓扑管理模块负责对链路发现模块发现的交换机之间的链路进行管理，形成网络空间的拓扑结构，并通过相应的编程接口向外提供自己保存的拓扑信息。下面对网络拓扑发现过程进行简单描述。

在控制器与交换机通过 OpenFlow 协议建立连接之后，控制器会通过 FeatureRequest 消息向交换机询问其状态信息，交换机在收到此消息之后，按照协议规定，必须向控制器发送自己的状态作息。此时控制器得到的是交换机的基本信息，并没有得到网络的拓扑结构信息。为了获取网络的拓扑信息，控制器的链路发现管理模块会周期性地向与之相连的交换机发送携带 LLDP(Link Layer Discovery Protocol)报文的 Packet-Out 消息，当交换机收到来自控制器的 LLDP 报文后会将其进行广播发送，当与之相连的交换机收到 LLDP 报文后会通过 Packet-In 消息返回给控制器。通过该过程，链路发现模块就会知道这两个交换机之间存在链接，并把对应的交换机的数据面 ID 和端口号记录下来，传送给拓扑管理模块。将该过程扩展至整个网络，控制器利用这种方式来获取整个网络的拓扑结构。

本文通过拓扑管理模块得到整个网络的拓扑结构信息，并通过控制器提供的编程接口得到网络中的交换机和链路的状态信息，把它们有效地结合以达到对网络路径异常进行监控的目的。网络拓扑结构管理模块会根据所得到的网络资源信息构建网络拓扑图 $G(V,E)$，其中结点集合 V 表示网络中的所有交换机，边集合 E 表示相对应的链路。

2.2　路径延迟测量模块

由于网络拓扑管理模块和路径异常监控模块已经得到了整个网络的交换机和网络拓扑情况，为了实现对网络路径的异常监控，首先需要得到路径的延迟情况。本文采用了一种主动的测量方式来对路径的延迟情况进行监控，控制器根据以下步骤得到路径的信息。首先，控制器预先在被测量路径的所有交换机上安装匹配探测数据包的转发规则，这些规则保证上一个交换机会把探测数据包转发给下一个交换机，这条路径上的最后一个交换机会触发 Packet-In 消息，并把探测数据包转发给控制器。然后，控制器会把携带当前系统时间戳的探测数据包发送到被测路径上的第一个交换机上，根据预先下发到交换机上的匹配规则，探测数据包会沿着被测路径转发到最近一个交换机上，再根据最后一个交换机返回的 Packet-In 消息得到此数据包。最后，控制器利用发送和接收探测数据包的时间差以及控制器到交换机之间的 Echo 消息往返时延计算路径的延迟，并把得到的路径延迟信息返回给路径异常监控模块。路径 S1－S2－S3－S4 的延迟测量过程如图 2 所示。首先，控制器把探测数据包的转发规则下发到对应的交换机上。接着，控制器发出携带时间戳 t_1 的探测数据包至交换机 S1，随后被 S4 用 Packet-In 消息返回，控制器收到此数据包的时间为 t_2。控制器到 S1 和 S4 的往返时间分别为 T_3 和 T_4，则路径 S1－S2－S3－S4 的延迟时间为：

$$T_{S1-S4}=(t_2-t_1)-\frac{T_3+T_4}{2}$$

本方法与现存的路径延迟测量方法相比有以下几个方面的优势：

1）利用 SDN 网络固有的控制数据包，既不需要对交换机的硬件进行任何修改，也不需要访问终端用户；

2）能够测量出任意两个 OpenFlow 交换机之间的路径延迟；

3）在控制器上计算路径的延迟时不需要考虑系统时钟同步的问题，可以对被测量的路径延迟进行集中的计算。

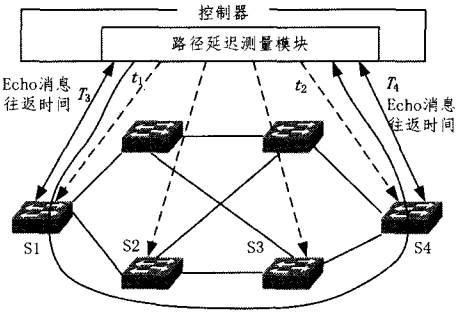

图 2　路径延迟测量过程

2.3　路径可用带宽测量模块

通过网络拓扑管理模块，路径异常监控模块已经获得了网络拓扑中各链路的基本信息，例如网络拓扑中各链路的带宽容量等。本文采用一种被动的测量方式来获得路径的可用带宽。首先，控制器通过 OpenFlow 协议中的PortStats-Reqest 消息以时间周期 T 询问交换机端口计数器中和 t 时刻的字节数 $N(t)$。然后，控制器根据一个周期内端口计数器字节数的变化情况计算出目前此链路的已用带宽 B：

$$B=\frac{N(t)-N(t-T)}{T}$$

控制器利用链路的带宽容量 C(Capacity) 以及此链路和已用带宽计算出对应链路的可用带宽 AB(Available Bandwidth)：

$$AB=C-B$$

最后，控制器根据给定路径上每条链路的可用带宽计算出整个路径的可用带宽 ABP(Path Available Bandwidth)：

$$ABP=\min_{AB\in P}AB$$

这种链路可用带宽的测量方法虽然是一种被动的测量方法，但是此方法的采样数据样本量较小，即使在高流量的网络中进行数据采样，其给网络来带的负载也是可以忽略的；另外，此方法是一种轮询的采样方式，与传统的采样方式相比，它可以根据需要对采样周期进行调整，对网络路径可用带宽的测量更具有灵活性。在进行数据采集时，控制器只需要相关路径上的交换机的端口信息，这在很大程度上减小了程序的开销。此方法可以测量出网络中任意路径的可用带宽，只要给定路径，控制器就能根据网络拓扑结构找出路径上对应的所有的交换机和链路，然后即可进行路径的可用带宽的测量工作。

在现有的工作中，研究者往往通过各种方式尽可能准确地得到网络中链路和路径的性能参数[13]（例如，链路的带宽容量或者链路的已用带宽），但是很少有研究者把测量链路带宽的工作集中到链路可用带宽的测量上，本文利用 SDN 网络架构的优势，通过计算链路可用带宽的方式，完成对路径的可用带宽的测量工作，并从不同的角度来反映网络路径的运行状态。

2.4　路径异常监控模块

首先，路径异常监控模块根据网络拓扑模块得到网络中所有交换机和链路的信息，然后根据路径可用带宽测量模块得到给定路径的可用带宽，并根据网络路径的带宽容量和可用带宽计算出路径的带宽剩余率：

带宽剩余率＝可用带宽/带宽容量

其中，路径的带宽容量为路径上所有链路的带宽容量的最小值，路径的可用带宽为路径上所有链路的可用带宽的最小值。路径带宽剩余率能比较真实地反映出网络路径的带宽使用情况。为了准确实现对网络路径的异常监控，路径异常监控模块需要周期性地从路径的可用带宽测量模块得到路径的可用带宽信息。路径可用带宽测量模块也恰好可以满足异常监控模块的这一需要，保证路径异常监控模块能够得到路径的带宽剩余率。

图 3 示出了异常监控模块的工作流程，当路径异常监控模块根据预先设定的阈值情况发现路径的带宽剩余率小于特定阈值时，异常测量模块就会调用路径延迟测量模块对这一路径的延迟进行测量；如果延迟测量模块得到的延迟测量结果大于某一阈值范围，则此路径出现异常，路径异常测量模块会立刻向控制器的负载均衡模块和防火墙模块发出报警信号，至于控制器如果何处理路径异常报警信号，则是后续的研究内容，在此不做赘述；如果路径的延迟测量结果并没有达到系统所设定的阈值范围，则路径异常监控模块不会产生报警信号，继续进行路径的监控任务，防止网络在运行过程当中产生错误识别的情况。

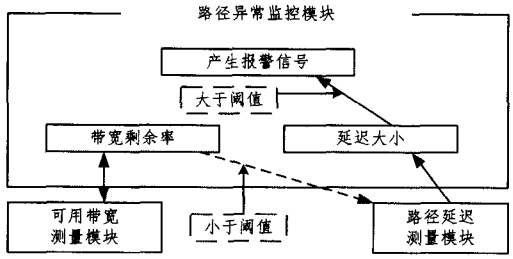

图 3 路径异常监控模块工作流程

3 实验结果与分析

本文把网络路径异常监控方法的 4 个模块作为 Floodlight 控制器的 4 个模块进行实现，并利用 Mininet 提供的接口进行自定义仿真网络拓扑的设计，构建如图 4 所示的网络拓扑结构。同时通过 D-ITG[10] 工具产生网络背景流量来模拟真实的网络流量，实现路径的拥塞，使路径的运行状态出现异常，实现复杂网络条件下的路径异常监控实验，并从以下 4 个方面来评估网络路径异常监控方法的准确性和有效性：

1)网络拓扑管理模块运行的准确性；

2)路径延迟测量结果的准确性；

3)路径可用带宽测量结果的准确性；

4)路径异常监控结果的准确性。

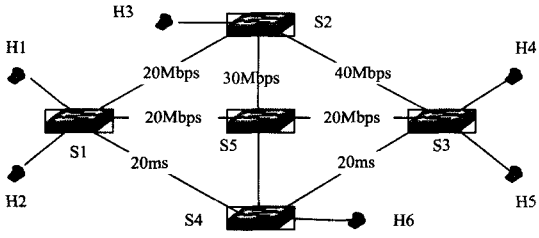

图 4 实验网络拓扑结构及链路性能信息

3.1 获取网络拓扑结构

网络拓扑管理模块通过 Floodlight 控制器向外提供的编程接口来获取网络拓扑结构，其准确性主要依赖于控制器获取网络拓扑结构的准确性。为了验证网络拓扑模块能够准确及时地得到网络拓扑结构的信息，通过增加和删除网络中的链路来观测拓扑管理模块中的拓扑变化情况。

表 1 中的数据为开始时的网络拓扑情况，0 表示交换机之间不存在链路，1 表示交换机之间存在链路。首先，用 Mininet 操作命令删除 S1 与 S5、S3 与 S5 之间的链路，再次查看网络拓扑管理模块中的拓扑信息。表 2 列出了断开后的网络拓扑状况，从表中可以看出两条链路已经成功从拓扑结构中删除。然后再用命令把去掉的链路添加上，再次获取网络拓扑情况，结果与表 1 列出的拓扑作息完全相同。

表 1 初始网络拓扑结构

交换机	S1	S2	S3	S4	S5
S1	0	1	0	1	1
S2	1	0	1	0	1
S3	0	1	0	1	1
S4	1	0	1	0	1
S5	1	1	1	1	0

表 2 断开链路后的网络拓扑结构

交换机	S1	S2	S3	S4	S5
S1	0	1	0	1	0
S2	1	0	1	0	1
S3	0	1	0	1	0
S4	1	0	1	0	1
S5	0	1	0	1	0

实验表明，控制器的链路发现模块和拓扑管理模块能够正确地管理网络的拓扑结构，链路发现模块利用链路层发现协议，周期性地向与相连的交换机发送数据报文进行链路发现，并将通过测量得到的链路信息结果发送给网络拓扑管理模块，拓扑管理模块根据链路发现模块得到的信息，结合自身的算法对网络的拓扑结构进行管理。本方法中的网络拓扑管理模块也能够通过控制器的编程接口正确地获取网络中拓扑结构的信息。

3.2 路径延迟测量

当路径异常监控模块在监控中发现路径传输过程中存在异常情况(如路径拥塞、可用带宽较低等)时，路径异常监控模块就会把此路径的信息传送到路径延迟测量模块。当给定一条路径，要对路径的延迟进行测量时，首先，路径测量模块会根据路径异常监控模块提供的路径信息来判别路径上的起始和终点交换机，也就是路径的起点和终点，然后下发特定的转发规则到路径上的每一台交换机，旨在引导探测数据包按照下发的规则沿着路径进行转发。接着，路径延迟测量模块会沿着路径的传输方向发送探测包到路径的第一个交换机上，并同时触发 Echo 消息到路径的起始和终点交换机上，Echo 消息的作用是测量控制器到交换机之间的路径延迟，控制器和交换机之间的链接在 OpenFlow 协议中被称为安全通道，在消息传递的过程中也会存在时间延迟。为了验证路径延迟测量模块的有效性，在如图 4 所示的拓扑结构中根据 Mininet 提供的编程接口预先设定了 S1－S4 之间的延迟时间 20ms 和 S4－S3 之间的链路延迟 40ms，然后分别用路径延迟测量模块测量出路径 S1－S4 和 S1－S4－S3 之间的延迟，同时手动地从主机 H1 执行 ping 命令，分别 ping 主机 H6 和 H5 来获取 ping 方法的延迟测量结果，图 5 和图 6 分别示出了两个路径延迟测量模块和 ping 命令得到的延迟结果。

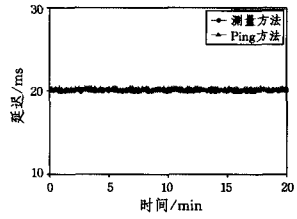

图 5 S1－S4 路径延迟结果

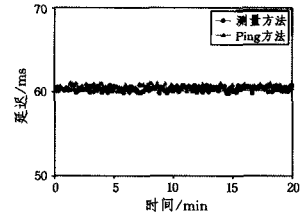

图 6 S1－S4－S3 路径延迟结果

从图中展示的测量结果可以得出，路径延迟测量模块能够比较准确地进行路径的延迟测量工作，通过预先向被测量路径上的交换机下发探测数据包的转发规则并触发 Packet-In 消息的过程，路径延迟测量模块能够测量出任意网络可达的两个交换机之间的路径延迟，并且具有毫秒级的测量水平，具有足够的精度对路径的延迟进行监控。

3.3 路径可用带宽测量

当路径异常监控模块给定一条路径且需要测量出该路径

的可用带宽时，首先，路径异常监控模块会把此路径上所有的交换机和链路信息传输给可用带宽测量模块；然后，可用带宽测量模块会根据路径异常监控模块提供的路径信息计算出路径的带宽容量；接着，轮询路径上所有交换机上端口计数器的统计信息；最后，根据公式计算出此路径的可用带宽并返回给路径异常监控模块。图 4 示出了设计和测量可用带宽的网络拓扑结构，其中 S1－S2－S3 组成了经典的 Y 型网络拓扑。为了验证路径可用带宽测量模块结果的准确性，用 D-ITG 工具设计了两种背景流量方案，一种是持续的网络背景流量方案，另一种是变化的背景流量方案。

持续背景流量：在此方案中，用 D-ITG 工具设计了如下的流量发送过程。首先，H1 向 H5 发送 UDP 流量，发送速率为 8Mbps，持续时间为 400s，200s 后再启动 H2 以同样的速率向 H5 发送 UDP 流量，持续时间为 200s。接着，中断 H1 和 H2 的流量发送，再用 H3 以 10Mbps 的速率向 H4 发送 UDP 流量直到结束，200s 后启动 H1 以 12Mbps 的速率向 H5 发送 UDP 流量直到结束。图 7 示出了可用带宽测量模块得到的测量结果，从图中可以看出，测量模块能比较真实地测量出路径的可用带宽。

变化背景流量：在此方案中，利用 D-ITG 工具按正态分布产生网络测量，沿着 S1－S2－S3 路径由 H1 发向 H5。对正态分布的参数进行了设定，平均值设置为 12，方差设置为 2.5，按照这样的分布向网络中动态地注入背景流量。图 8 示出了路径可用带宽测量模块得到的测量结果，从图中展示的信息可以得到，在流量动态变化的过程中，测量结果中的局部虽然有一定的测量误差，但是整体上能够反映出路径可用带宽的变化情况。

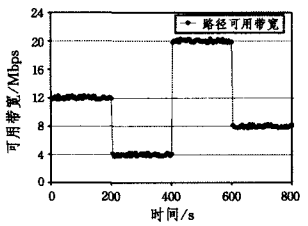

图 7　持续速率测量结果

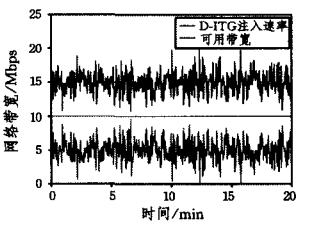

图 8　变化速率测量结果

3.4　路径异常监控结果

由上文所知，网络拓扑管理模块、路径延迟测量模块和路径可用带宽测量模块都能很好地实现本模块的功能。路径异常监控模块根据网络拓扑管理模块得到网络的拓扑结构、交换机以及链路的基本信息，并通过路径可用带宽模块得到路径的可用带宽来计算路径的带宽剩余率，如果路径带宽剩余率小于某一阈值（阈值设定的参考标准可以按照对网络的测量结果进行统计分析；在实验中，根据路径出现异常的情况对路径的带宽剩余率和延迟进行统计，把带宽剩余率的阈值设定为 20%），异常监控模块就会调用通知延迟测量模块对路径的延迟情况进行测量。实验过程中，把 H1 到 H5 的发送速率从 0Mbps 增长到 21Mbps，在进行端口计数器采样的同时也利用路径延迟测量模块对此路径进行延迟测量。

实验表明，当路径带宽剩余率低于 20%时，路径的延迟开始迅速增长，路径开始出现拥塞的情况，此路径异常监控模块就会产生报警信号到控制器中的相关功能模块。

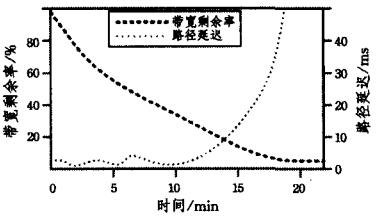

图 9　带宽剩余率与路径延迟的变化关系

结束语　网络的使用已经渗透到生活的各个领域，网络规模也在不断壮大，结构层次越来越复杂，因此迫切需要一种高效的网络路径异常监控的方法对网络路径进行有效的异常监控，以保证网络的正常运行。本文利用网络测量中主动测量与被动测量相结合的方式，提出了一种 SDN 网络中路径异常监控的方法，利用被动监控、主动检测的思想提高了路径异常监控的准确性。实验表明，4 个模块在网络运行时能够各自完成对应的功能并且能有效地进行协调工作，准确地对路径的异常情况进行监控与检测。然而，在实验过程中使用的仍然是仿真的网络拓扑结构，虽然表现出了很好的路径异常监控效果，但是此方法还没有在真实的大规模网络中得到应用。接下来的工作应把此方法应用于真实的网络中作为路径异常监控的基本方法，以验证其有效性和准确性。

参 考 文 献

[1] KREUTZ D, RAMOS F M V, VERÍSSIMO P E, et al. Software-Defined Networking: A Comprehensive Survey[J]. Proceedings of the IEEE, 2014, 103(1): 10-13.

[2] OpenFlow Networking Foundation[OL]. http://www.opennetworking.org.

[3] YU C, LUMEZANU C, ZHANG Y, et al. FlowSense: Monitoring Network Utilization with Zero Measurement Cost[C]// International Conference on Passive and Active Measurement. 2013: 575-579.

[4] CHOWDHURY S R, BARI M F, AHMED R, et al. PayLess: A low cost network monitoring framework for Software Defined Networks[C]// 2014 IEEE/IFIP Network Operations and Management Symposium. 2014: 1-9.

[5] RAUMER D, SCHWAIGHOFER L, CARLE G. MonSamp: A distributed SDN application for QoS monitoring[C]// 2014 Federated Conference on Computer Science and Information Systems (FedCSIS). IEEE, 2014: 961-968.

[6] VAN ADRICHEM N L M, DOERR C, KUIPERS F A. OpenNetMon: Network monitoring in OpenFlow Software-Defined Networks[C]// 2014 IEEE/IFIP Network Operations and Management Symposium. 2014: 1-8.

[7] BABIARZ J. A Two-Way Active Measurement Protocol (TWAMP)[OL]. http://tools.ietf.org/html/rfc5357.

[8] DONNELLY S, MCGREGOR T. Design principles for accurate passive measurement[M]// Passive and Active Measurement Workshop, 2000.

[9] Floodlight Controller[OL].https://floodlight.atlassian.net/wiki.

[10] BOTTA A,DAINOTTI A,PESCAPÉ A. A tool for the generation of realistic network workload for emerging networking scenarios[J]. Computer Networks (Elsevier), 2012, 56(15): 3531-3547.

[11] CAI Z P. Study on network measurement technology, model and algorithm based on active and passive measurement [D]. Changsha: National University of Defense Technology, 2005. (in Chinese)
蔡志平. 基于主动和被动测量的网络测量技术、模型和算法研究[D]. 长沙:国防科学技术大学,2005.

[12] CHENG G, WANG Y X, HU Y F, et al. Network Link Fault Diagnosis Based on OpenFlow[J]. Journal of Beijing University of Posts and Telecommunications, 2015, 38(5): 42-46. (in Chinese)
程光,王玉祥,胡一非,等. 基于 OpenFlow 的链路故障诊断方法[J]. 北京邮电大学学报,2015,38(5):42-46.

[13] CAI Z P, LIU F, ZHAO W T, et al. Network measurement deployment model and its optimization algorithm[J]. Journal of Software, 2008, 19(2): 419-431. (in Chinese)
蔡志平,刘芳,赵文涛,等. 网络测量部署模型及其优化算法[J]. 软件学报,2008,19(2):419-431.

(上接第 20 页)

[6] CHEN Y R. Research on User Behavior Authentication and Security Control in Cloud Computing[D]. Beijing: University of Science and Technology Beijing, 2012. (in Chinese)
陈亚睿. 云计算环境下用户行为认证与安全控制研究[D]. 北京:北京科技大学,2012.

[7] FENG D G, ZHANG M, ZHANG Y, et al. Study on Cloud Computing Security[J]. Journal of Software, 2011, 22(1): 71-83. (in Chinese)
冯登国,张敏,张妍,等. 云计算安全研究[J]. 软件学报,2011,22(1):71-83.

[8] LUNA G J, LANGENBERG R, SURI N. Benchmarking cloud security level agreements using quantitative policy trees[C]// ACM Workshop on Cloud Computing Security Workshop. 2012: 103-112.

[9] SHETTY S. Auditing and Analysis of Network Traffic in Cloud Environment[C]// IEEE Ninth World Congress on Services. 2013: 235-258.

[10] BIRNBAUM Z, LIU B, DOLGIKH A, et al. Cloud Security Auditing Based on Behavioral Modeling[J]. International Journal of Business Process Integration & Management, 2013, 7(2): 268-273.

[11] GANJALI A, LIE D. Auditing Cloud Administrators Using Information Flow Tracking[C]// Proceedings of the 7th ACM Workshop on Scalable Trusted Computing. 2012: 79-84.

[12] WANG X, ZHANG J, WANG M, et al. CDCAS: A Novel Cloud Data Center Security Auditing System[C]// IEEE International Conference on Services Computing. IEEE, 2014: 605-612.

[13] BIRK D, WEGENER C. Technical Issues of Forensic Investigations in Cloud Computing Environments[C]// IEEE Sixth International Workshop on Systematic Approaches To Digital Forensic Engineering. IEEE, 2011: 1-10.

[14] ATENIESE G, BURNS R, CURTMOLA R, et al. Provable data possession at untrusted stores[C]// ACM Conference on Computer and Communications Security. ACM, 2007: 598-609.

[15] ATENIESE G, PIETRO R D, MANCINI L V, et al. Scalable and Efficient Provable Data Possession[C]// Proceedings of the 4th International Conference on Security and Privacy in Communication Networks. ACM, 2008: 1-10.

[16] JUELS A, KALISKI B S. Pors: proofs of retrievability for large files[C]// ACM Conference on Computer and Communications Security. ACM, 2007: 584-597.

[17] WANG C, CHOW S S M, WANG Q, et al. Privacy-Preserving Public Auditing for Secure Cloud Storage[J]. IEEE Transactions on Computers, 2013, 2009(2): 362-375.

[18] WANG Q, WANG C, LI J, et al. Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing[C]// European Conference on Research in Computer Security. Springer-Verlag, 2009: 355-370.

[19] ZHU Y, WANG H, HU Z, et al. Dynamic audit services for integrity verification of outsourced storages in clouds[C]// Proc. of the 2011 ACM Symposium on Applied Computing (SAC). 2011: 1550-1557.

[20] KAI H, CHUANHE H, JINHAI W, et al. An Efficient Public Batch Auditing Protocol for Data Security in Multi-cloud Storage[C]// Chinagrid Conference. IEEE Computer Society, 2013: 51-56.

[21] YANG K, JIA X. An Efficient and Secure Dynamic Auditing Protocol for Data Storage in Cloud Computing[J]. IEEE Transactions on Parallel & Distributed Systems, 2013, 24(9): 1717-1726.

[22] HE K, HUANG C H, WANG X M, et al. Aggregated privacy-preserving auditing for cloud data integrity[J]. Journal on Communications, 2015, 36(10): 119-132. (in Chinese)
何凯,黄传河,王小毛,等. 云存储中数据完整性的聚合盲审计方法[J]. 通信学报,2015,36(10):119-132.

[23] BLEIKERTZ S, SCHUNTER M. Security audits of multi-tier virtual infrastructures in public infrastructure clouds[C]// Proceedings of the 2010 ACM Workshop on Cloud Computing Security Workshop. New York: ACM Press, 2010: 93-102.

[24] BLEIKERTZ S. Automated Security Analysis of Infrastructure Clouds[J]. Institutt for Telematikk, 2010, 18: 113-122.

[25] DOELITZSCHER F, FISHER C, MOSKAL D, et al. Validating Cloud Infrastructure Changes by Cloud Audits[C]// Services. 2012: 377-384.

[26] DOELITZSCHER F, REICH C, KNAHL M, et al. An agent based business aware incident detection system for cloud environments[J]. Journal of Cloud Computing, 2012, 1(1): 1-19.

[27] MADI T, MAJUMDAR S, WANG Y, et al. Auditing Security Compliance of the Virtualized Infrastructure in the Cloud: Application to OpenStack[C]// ACM Conference on Data and Application Security and Privacy. ACM, 2016: 195-206.