基于可编程交换机的网内灰色故障检测技术研究进展

刘宏岩[1],张 栋[2],吴春明[1*]
(1. 浙江大学计算机科学与技术学院,浙江杭州 310063;2. 福州大学计算机与大数据学院,福建福州 350108)

摘 要: 灰色故障是指对生产网络产生细微影响的交换机故障. 然而,当这些轻微故障相互叠加或与新增故障叠加时,可能会导致整个生产网络的瘫痪. 因此,检测灰色故障对生产网络的稳定性至关重要. 传统解决方案关注的是在控制平面收集数据平面交换机中的流记录,并对其进行处理以检测灰色故障. 然而,此类解决方案存在着不足:(1)缓存和处理大量的流记录会引入庞大的资源开销;(2)较高的检测时延无法保证灰色故障检测的时效性. 近年来,可编程交换机的出现为灰色故障检测技术带来了新机遇:网络管理员可以将灰色故障检测算法部署运行至可编程交换机的线速 ASIC 流水线上,从而支持低开销、低时延、高精度的网内灰色故障检测技术. 本文针对基于可编程交换机的网内灰色故障检测技术进行综述,在对灰色故障的概念、普遍性及对生产网络的危害进行描述的基础上,分析与讨论了现有基于可编程交换机的网内灰色故障检测技术的研究现状与进展,详细介绍每项技术的工作原理及流程,搭建真实的实验平台评估各项技术的检测指标,在文末指出了现有技术所面临的问题与挑战.

关键词: 灰色故障检测;可编程交换机;网内计算;网络测量;数据报丢失;数据中心网络
基金项目: 浙江省“尖兵”“领雁”研发攻关计划项目(No.2024C01066)
中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112(2024)10-3613-10
电子学报 URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20240199

Empowering In-Network Gray Failure Detection with Programmable Switches

LIU Hong-yan[1], ZHANG Dong[2], WU Chun-ming[1*]
(1. *College of Computer Science and Technology*, *Zhejiang University*, *Hangzhou*, *Zhejiang* 310063, *China*;
2. *College of Computer and Data Science*, *Fuzhou University*, *Fuzhou*, *Fujian* 350108, *China*)

Abstract: Gray failures are micro switch malfunctions that have a subtle impact on production networks. However, when these micro malfunctions are superimposed on each other or on a new malfunction, they can lead to paralysis of production networks. Thus, the detection of gray failures is essential to the stability of production networks. Prior methods focus on using the control plane to collect flow records from data plane switches and process them to detect packet loss. However, they fall short due to (1) their high resource overhead of handling with massive flow records and (2) non-trivial delays that result in out-of-date failure detection. Recently, the emergence of programmable switches provides a promising alternative solution: the detection of gray failures can be offloaded to line-rate switch ASIC pipelines, enabling low-cost, low-latency, and high-accuracy in-network gray failure detection. This paper presents an illustrative survey of programmable switch-assisted techniques in in-network gray failure detection. First, we describe the concept of gray failures, their prevalence, and their impact to production networks. Second, we analyze and discuss the characteristics of state-of-the-art gray failures detection techniques built on programmable switches. Third, we illustrate the principle and workflow of each detection technique. Fourth, we conduct a real-world testbed to evaluate the metrics of each detection technique. Finally, we highlight the problems and challenges faced by existing techniques.

Key words: gray failure detection; programmable switches; in-network computing; network measurement; packet loss; datacenter networks

收稿日期:2024-02-29;修回日期:2024-08-20;责任编辑:宋甲英
*通讯作者:吴春明

Foundation Item(s): The “Pioneer” and “Leading Goose” Research and Development Program of Zhejiang (No.2024C01066)

1 引言

在过去的二十年中,网络管理员一直受到灰色故障的挑战[1,2]. 其中,灰色故障指的是细微的交换机故障(如配置漏洞),导致网络设备转发的部分流量丢失[2]. 单个灰色故障对网络的影响可能是轻微的,但是当大量的灰色故障相互叠加,或者与一个新的较大故障叠加时,将会导致生产网络瘫痪. 因此,网络管理员希望快速地检测网络中的灰色故障,以提高网络的稳定性和可用性. 在包括网络服务提供商网络(Internet Service Provider network,ISP)和数据中心网络的生产网络中,灰色故障普遍存在[1,3]. 由于灰色故障表现为网络设备的间歇性或局部丢包,网络管理员通常通过检测数据报丢失进而检测灰色故障[1,4,5].

为检测数据报丢失,一个草创的方案是在控制平面调用交换机操作系统提供的接口,收集并比较上游交换机与下游交换机的端口数据报计数. 然而,此类解决方案无法具体确定丢失的数据报及其流量特征(如五元组),不利于后续的灰色故障定位与消除. 例如,基于丢失数据报的五元组信息,网络管理员可以快速定位交换机内部具体的数据报处理路径,从中排查出此类数据报丢失的原因,将灰色故障的排查范围从全部处理路径缩小到部分处理路径,提高灰色故障定位及消除效率.

为了解决上述缺陷,现有主流的灰色故障检测技术基于控制平面提供的全局视野,以捕捉丢失数据报的详细信息,即“以控制平面为中心的检测技术”. 具体来说,这些技术可分为以下两类. 第一类检测技术将进入交换机的全部数据报镜像到控制平面,从而准确地检测是否以及那些有数据报丢失[6,7]. 然而,由于交换机在运行时的吞吐量可以达到Tbps级别(如Intel Tofino交换机支持6.4 Tbps的吞吐量[8]),将所有流量镜像到控制平面将耗费大量网络带宽和控制平面资源来处理数据报. 因此,基于镜像的检测技术虽然具有高检测精度的优势但并不实用,具有高开销的缺点.

为减少基于镜像检测技术带来的资源开销,第二类检测技术选择对流量进行采样,例如从每五个数据报中选择一个作为样本汇报到控制平面,大大减少了发送到控制平面的数据量[9,10]. 然而,数据报采样失去了对流量的全局可见性,无法精确地捕捉到所有丢失的数据报. 因此,基于采样的检测技术虽然减少了检测开销,但也牺牲了检测精度.

除此之外,以上两种技术都需要将数据报从数据平面送往控制平面. 控制平面分析收到的数据报,检测是否有数据报丢失,随后做出故障恢复决策. 然而,由于控制平面与数据平面之间存在着较高的传输时延[11,12],导致这一整个过程的检测时延较高,进而影响故障恢复决策的时效性. 综上所述,主流灰色故障检测技术无法提供同时满足低开销、低时延、高精度的灰色故障检测.

近年来,学术界研发出了可编程交换机. 可编程交换机的出现使网络管理员在控制和管理数据平面方面获得了前所未有的可编程性[13]. 具体来说,与功能固化的传统交换机相比,可编程交换机具有两个极具吸引力的特点. 首先,网络管理员可以利用高级编程语言(如P4[14])动态按需变更运行在可编程交换机上的数据报处理逻辑. 其次,可编程交换机可确保Tbps级别的吞吐量和微秒级的包处理时延. 利用上述优势,网络管理员可以将一些传统上由控制平面完成的网络管理操作卸载到可编程交换机上.

在此背景下,学术界注意到可编程交换机带来的机遇,提出多种基于可编程交换机的网内灰色故障检测技术. 确切地说,此类检测技术直接在可编程交换机上实现灰色故障检测算法,无须将数据报发送到控制平面,即网内检测,降低了检测时延又减少了因数据报传输带来的资源开销. 在线速处理性能的加持下,网内检测技术能够在不牺牲检测精度的前提下处理Tbps级别的流量.

表1对比了基于可编程交换机的网内灰色故障检测技术与两类传统技术,即基于镜像的检测技术和基于采样的检测技术. 相比于基于镜像的检测技术,基于可编程交换机的网内灰色故障检测技术降低了检测开销和时延. 相比于基于采样的检测技术,基于可编程交换机的网内灰色故障检测技术提高了检测精度且降低了检测时延.

表1 基于可编程交换机的网内灰色故障检测技术与传统技术的对比

技术特点	部署位置	时延	精度	开销
基于镜像	控制平面	高	高	高
基于采样	控制平面	高	低	低
基于可编程交换机	数据平面	低	高	低

尽管已有一些针对灰色故障检测或可编程网络的综述文献[8,15~17],但它们尚未对近期提出的、基于可编程交换机的网内灰色故障检测技术进行全面综述. 为此,本文对近十年、基于可编程交换机的网内灰色故障检测技术进行全面调研,从而弥补这一空白. 首先,阐述了灰色故障的概念及其对生产网络的危害. 其次,解

释了为什么灰色故障检测技术从传统的“以控制平面为中心”转向“网内检测”. 再次,阐述了基于可编程交换机的网内灰色故障检测技术的工作流程,并对每种相关技术进行讨论、分类,逐一分析它们的优缺点. 再次,通过搭建真实实验测试平台,对每种灰色故障检测技术进行评估. 这有助于网络管理员深入理解每种技术的特性,在实际生产环境中选择合适的技术. 最后,分析现有基于可编程交换机的网内灰色故障检测技术存在的问题,探讨可能的发展方向.

2　生产网络中的灰色故障

2.1　灰色故障

现代生产网络通常包含数千至数万个数据平面交换机[18]. 在这种错综复杂的网络布局中,运行着诸如网络搜索、在线支付和股票交易等多种互联网应用. 这些应用对网络的可用性提出了严格要求:最大程度地减少网络停机,保障服务的连续性和质量[19]. 然而,在大规模生产网络中,由于人为配置失误、软件漏洞和硬件故障等多种因素,网络中时常出现难以察觉的微小故障,使网络设备转发的部分流量丢失,即灰色故障[2,20]. 例如,网络管理员在开发数据平面程序时,针对某类正常流量的处理逻辑存在开发漏洞,将其导入对恶意流量的处理路径,导致属于此类流量的数据报全部被丢弃[21]. 虽然单个灰色故障对网络的影响看似微不足道,但多个灰色故障的相互作用,或与一起重大故障叠加之后,能够对网络造成破坏性的影响. 因此,灰色故障对于维护网络的高可用性构成了一大挑战.

如图1所示,灰色故障引起了网络管理员的极大关注,频繁出现在各类生产网络中. 以往的研究报告显示,数据中心网络中每年有大约平均40至80个交换机遭受灰色故障的影响,导致的它们的数据报丢包率超过50%[22]. 数据报丢失会对网络应用的性能造成严重的影响,例如增加用户请求的处理时延. 在ISP网络中,灰色故障同样频繁发生. 最新的研究表明,超过90%的网络管理员表示曾遭遇过灰色故障[2]. 这些故障不仅引发了大量数据报丢失,还会影响流量路由,甚至在ISP网络中形成了路由黑洞. 更糟的是,由于灰色故障的隐蔽性,它们极难被及时检测到. 网络管理员可能需要花费数小时甚至数天,在某些极端情况下达数月的时间来定位和恢复这些故障[5].

综上可得:(1)灰色故障普遍存在于现代生产网络中;(2)灰色故障会对网络造成严重影响,比如引发丢包和路由黑洞;(3)对大多数网络管理员而言,检测灰色故障是一个耗时且困难的任务.

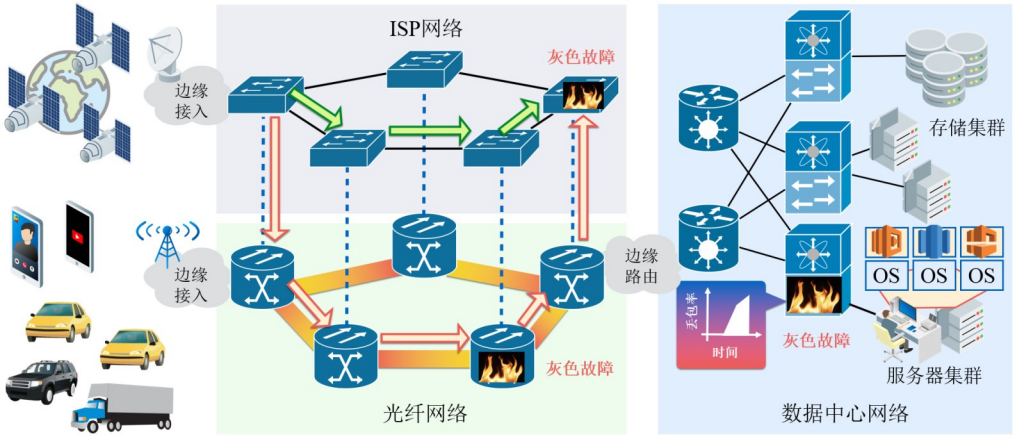

图1　灰色故障普遍存在于生产网络中,导致大量数据报丢失

2.2　灰色故障检测

近年来,学术界提出了许多灰色故障检测技术. 在理想情况下,网络管理员希望所提出的灰色故障检测技术同时满足低开销、低时延、高精度三种特性. 具体来说,低开销指的是检测技术应避免消耗大量的网络资源,影响其他应用的正常运行. 低时延指的是检测技术应尽可能快地发现正在发生的故障,最小化灰色故障带来的影响. 高精度指的是检测技术应准确地对引发灰色故障的交换机进行定位.

在研发同时满足上述三点需求的技术方案之前,网络管理员首先要选定一个能够反映灰色故障的指标. 根据相关文献,是否存在数据报丢失是检测灰色故障的一项重要指标[2,20],原因包括以下两点. 首先,灰色故障的直接后果是数据报丢失,因为交换机功能故障将不可避免地干扰到数据报的处理过程. 其次,数据报丢失会直接影响到应用程序的性能(如增加网络延迟). 因此,通过监控和分析数据报的丢失情况,网络管理员可以有效地检测和识别网络中潜在的灰色故障.

3　以控制平面为中心的灰色故障检测技术

接下来,本文介绍目前主流的灰色故障检测技术,

即以控制平面为中心的灰色故障检测技术. 此类技术采用含有控制平面和数据平面的框架. 在数据平面,交换机将到达的数据报发送到控制平面. 接下来,控制平面基于收到的数据报检测是否有数据报丢失和灰色故障. 本文将现有的、以控制平面为中心的灰色故障检测技术归纳为两类.

3.1 基于镜像的灰色故障检测技术

首先,基于镜像的灰色故障检测技术将交换机接收到的数据报全部镜像到控制平面,由控制平面负责完成所有灰色故障检测算法[6,7]. 如图2所示,首先,基于镜像的灰色故障检测技术让每个交换机将所有到达的数据报镜像到控制平面. 接下来,控制平面会枚举每个交换机,并找出它的所有下游相邻交换机. 然后,控制平面汇总下游交换机收到的所有数据报,并与当前交换机镜像的数据报进行对比. 如果二者存在差异,就意味着发生了数据报丢失.

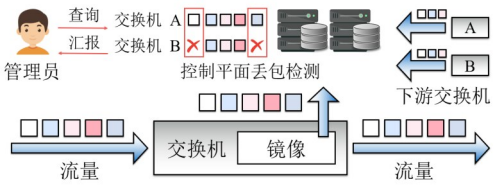

图2 基于镜像的灰色故障检测技术的工作流程

然而,基于镜像的灰色故障检测技术牺牲了检测时效性并具有较高开销. 首先,控制平面和数据平面是通过链路相连的,这些链路的带宽往往有限(约几十Gbps)[23]. 但是,数据平面交换机接收流量的速率接近几Tbps. 因此,流量镜像会导致平面间链路拥塞,带来了较高的检测时延. 其次,控制平面需要启用服务器集群来处理所有镜像数据报,引入了高昂的设备启用和维护开销.

3.2 基于采样的灰色故障检测技术

其次,如图3所示,基于采样的灰色故障检测技术提前在交换机上设置了一个采样率γ[9,10]. 交换机会根据这个采样率从到达的数据报中选取一部分作为样本,发送到控制平面. 在图3中,网络管理员将采样率γ设置为1:2,交换机就会从每两个到达的数据报中挑选一个发送到控制平面. 因此,基于采样的灰色故障检测技术减少了需要向控制平面发送的数据量.

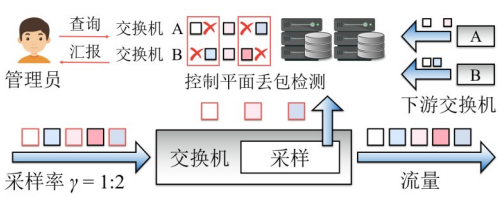

图3 基于采样的灰色故障检测技术的工作流程

然而,基于采样的灰色故障检测技术仍然存在着两个严重的局限性. 首先,此类技术仍无法实现低时延的灰色故障检测. 这是因为此类技术本质上还是以控制平面为中心进行灰色故障检测,受限于数据平面与控制平面间较高的通信时延[12]. 其次,基于采样的灰色故障检测技术会漏掉大部分数据报. 所以检测结果精度较低,产生误报.

4 基于可编程交换机的网内灰色故障检测技术

最近,可编程交换机的出现衍生出一类新的灰色故障检测技术,即基于可编程交换机的网内灰色故障检测技术. 此类技术将灰色故障检测操作从控制平面转移到了数据平面可编程交换机,实现低开销、低时延、高精度的灰色故障检测.

4.1 基于可编程交换机的网内灰色故障检测技术的基本框架

可编程交换机作为学术界中新兴的趋势,在灰色故障检测方面具备以下三方面优势. 首先,网络管理员可以利用高级编程语言动态按需变更运行在可编程交换机ASIC流水线上的灰色故障检测策略,具备较高的灵活性. 其次,一旦这些策略成功地部署在可编程交换机ASIC流水线上,可编程交换机就能以线速执行这些策略,实现了高性能,低时延的灰色故障检测. 第三,可编程交换机具有较高的性价比,其采购和维护成本对生产网络来说是可接受的[24]. 如图4所示,构建此类检测技术的工作流程主要包括以下三个步骤.

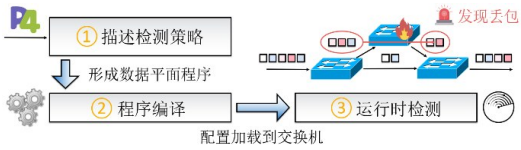

图4 构建基于可编程交换机的网内灰色故障检测技术的工作流程

(1)基于高级语言的灰色故障检测策略描述. 网络管理员通过高级编程语言(如P4)在数据平面程序中描述灰色故障检测策略. 每个数据平面程序包含多个匹配动作表(Match-Action Table,MAT)和一条控制流. 其中,MAT根据网络管理员指定的规则匹配数据报头部字段,并根据匹配结果执行相应的动作(如修改头部值). 控制流则规定了所有MAT的执行顺序. 通过精心地设计MAT和控制流,网络管理员可以在数据平面程序中实现任何灰色故障检测策略[25].

(2)面向可编程交换机的灰色故障检测策略加载. 网络管理员将编码有灰色故障检测策略的数据平面程序输入到交换机编译器,生成与可编程交换机兼容的硬件配置,最终加载到可编程交换机ASIC流水线上[26].

(3)运行时网内灰色故障检测. 在运行时,可编程交换机使用MAT和带状态组件(如寄存器)构建用于检测数据报丢失的数据结构和算法. 对于到达的流量,可编程交换机可以在线速率对其实施检测算法. 如果发现丢包,则会向控制平面发出警报. 由于这些数据结构和算法在不同技术中各不相同,在后续会对每种技术展开详细介绍.

4.2 基于可编程交换机的网内灰色故障检测技术的研究现状

目前已经有一些基于可编程交换机的网内灰色故障检测技术,本文选取了7项具有代表性的技术,按照它们发表的年份次序逐一进行介绍,阐述早期技术的不足及最新技术的动机与优势.

文献[27]针对基于采样的灰色故障检测技术精度不足的问题,提出了一个同时兼顾检测精度和检测开销的方案FlowRadar(FR). FR在可编程交换机ASIC流水线上维护一个哈希表,存储每条流的标识符(如五元组)及其数据报数量. 控制平面定期从数据平面获取哈希表,从中查询信息以完成灰色故障检测. 通过这种方式,FR减少了发送到控制平面的数据量,同时保留了每个交换机对流量的完整视图,实现了高精度.

实现FR的主要挑战是如何处理不同流之间的哈希冲突. 由于交换机的内存资源有限,FR只能在哈希表中定义少量计数器. 考虑到数据平面庞大的流数量,哈希冲突不可避免. 为此,FR在发生哈希冲突时,通过异或操作来编码冲突流的流标识符并累加它们的数据包数量. 此外,FR还会在每个计数器记录被哈希到该计数器的不同流数量. 随后,FR将整张哈希表送到控制平面. 由于异或操作的特性,控制平面只需要知道一条流的准确数据报数量(即寻找那些冲突流计数为1的计数器),就可以不断地解码出其他流的数据报数量.

然而,FR存在着几点局限性. 首先,FR仍然依赖控制平面进行数据解码与灰色故障检测,因此无法实时地检测灰色故障. 其次,FR需要频繁地(如每隔10 ms)收集哈希表. 即使是小型哈希表(如2^{16}个32字节计数器),仍然会占用不小的链路带宽(如几十Gbps[12]). 如果与普通流量共享带宽,容易导致链路拥塞,影响服务质量.

为了解决FR高带宽消耗的问题,文献[22]提出了一种轻量级技术LossRadar(LR). 相比FR在每台交换机中记录每条流的统计数据,LR仅记录丢失数据报的统计信息. 因为丢失数据报的数量远少于整体流量,LR大大提高了资源效率.

具体而言,LR的设计目标是在短时间内(几十毫秒)准确报告一个网络域(如一台交换机或一片局域网)内丢失数据报的具体信息(如五元组). LR在网络域的入口节点和出口节点上分别安装了一个轻量级模块,称为流量计. 流量计会测量通过它的流量,形成流量摘要. LR定期将两处的流量摘要报告给控制平面. 控制平面则对比入口节点与出口节点间流量摘要的差异获取网络域内丢失数据报的必要信息.

流量计基于可逆布隆过滤器实现,其内部每个计数器包含两个字段:流量标识符字段与数据报数量字段. 当数据报到达时,流量计首先提取其流标识符. 随后将流标识符输入一组哈希函数以选定若干计数器. 对于每个选定的计数器,流量计将到达数据报的流标识符与已存储的标识符字段进行异或编码,并将数据报数量字段加一.

然而,LR有两点局限性. 首先,LR仍然依赖控制平面进行丢包和灰色故障检测,导致毫秒级的检测时延. 其次,频繁地从交换机收集流量摘要需要消耗大量的交换机资源. 最近的研究表明[2],当网络丢包率达到0.1%时,LR需要消耗1.7倍的交换机内存来存储流量摘要. 在这种情况下,除非对交换机进行资源扩容,否则LR无法正常工作.

文献[28]提出了一项与FR、LR同期的检测技术Marple(MP). MP旨在利用可编程交换机的特性构建一组流量测量原语,网络管理员可以使用这些原语灵活地定制流量测量任务. 例如,网络管理员可以将全网流量划分为几组,每组对应到一个数据平面交换机. 然后在交换机中计算每组流量中每条流的数据报数量. 通过比较同一条流在不同组中的计算结果,检测数据报丢失和灰色故障. 随后,MP将用户输入的灰色故障检测任务编译成一条控制流. 这条控制流按顺序调用可编程交换机中的硬件单元来完成任务.

在MP提供的所有原语中,最难实现的原语是在交换机ASIC流水线上计算一组流量中每条流的数据报数量. 因为这项操作会消耗大量的交换机内存资源,甚至超过交换机的资源总量. 为此,MP设计了一个哈希表,在哈希表中维护若干条目. 每个条目绑定一条特定的流的数据报数量. 当发生哈希冲突时,即一条新流被哈希到记录旧流的条目,MP会将旧流的数据驱逐到控制平面的后备存储中,为新流腾出空间.

然而,MP会带来不小的带宽开销. 原因在于到达数据平面交换机的流量达Tbps级别,而MP提供的哈希表容量有限,导致大量哈希表条目在短时间内被驱逐到后备存储,对数据平面与控制平面之间的链路造成了巨大的带宽压力. 此外,类似于FR和LR,MP仍然依靠控制平面进行检测,因此也存在高检测时延的缺陷.

文献[29]提出了一种完全在数据平面实现灰色故障检测的技术BLink(BL). 与上述引入控制平面的技术相比,BL将检测时延缩短了一个量级. BL的核心思想

是:当TCP流遭遇灰色故障时,它们的数据报会丢失,随后会不断地重传丢失的数据报.因此,TCP流的重复重传行为成为了一个明显的检测信号.

基于这一思想,BL在可编程交换机ASIC流水线中检测这些重传信号.BL包含以下三部分设计.首先,BL只监控给定前缀中的部分活跃流,以减少对交换机资源的消耗.当某条活跃流被长时间监控或中断时,BL会将其替换为其他流.由于现代网络流量通常集中在少数几条活跃流,这使得BL在保持低开销的同时,能够覆盖大部分流量.其次,为避免受到噪声信号的影响,BL采用了一种智能检测策略:当某个目地前缀的大量活跃流失去连接(即发生超时重传的流数量超过阈值)时,BL就会判断该目地前缀处发生故障.第三,当检测到故障时,BL会迅速根据用户指定的故障修复策略修改即将发送数据报的下一跳路由.

BL网内检测的设计思路使它能够在亚毫秒级的时延内检测到灰色故障,同时满足高精度、低开销的要求,超越了之前的多项技术.然而,BL只关心影响活跃TCP流的灰色故障,无法覆盖到所有类型的灰色故障.比如,BL无法检测影响UDP流的灰色故障.但由于交换机资源有限,让BL支持所有类型的灰色故障也是不现实的.

文献[5]提出了一种与BL相似的检测技术NetSeer(NS).NS用于检测生产网络中的网络性能异常(Network Performance Anomaly,NPA).NPA在短时间内随机发生,成为大多数网络故障的诱因.NS观察到大部分NPA都源于数据报丢失和灰色故障.因此,NS尽快地检测数据报丢失和灰色故障,以便及时定位并消除NPA.

为此,NS设计了一个网络监控系统,在可编程交换机中检测数据报丢失、排队、乱序和暂停四种事件.为了精准地检测每种事件,NS在每个交换机上配置事件专属的检测逻辑.在运行时,NS首先根据检测逻辑筛选出经历事件的数据报.其次,NS按流级别对筛选出的数据报去重,只保留一条流的一个数据报.这样可以大幅减少向控制平面发送的数据量.随后,NS提取去重后每个数据报的关键信息(如五元组),进一步优化数据传输引入的带宽开销.最后,在发送数据时,NS不会依次发送每个数据报的关键信息,因为这样会产生大量的短报文,耗费控制平面的CPU资源.相反,NS将若干关键信息分批打包发送至控制平面.

其中,为了检测数据报丢失事件,NS考虑了两种情况:交换机内丢包与交换机外丢包.首先,交换机内丢包通常由ACL表、TTL值归零、超过MTU、拥塞等原因引起.因此,NS在交换机中插入了检测这些情况的匹配规则,以筛选经历交换机内丢包事件的数据报.其次,交换机外丢包通常由连接上下游交换机的链路故障引起.因此,NS在上游交换机创建一个环形缓冲区,用于缓存最近发送的数据报,并向每个发送的数据报增加序列号.当下游交换机收到带有不连续序列号的两个数据报时,向上游交换机发送数据报丢失警告.随后,上游交换机根据不连续序列号之间的差值在环形缓冲区中找到丢失的数据报.

然而,NS存在两个缺陷.首先,NS同样依赖控制平面进行灰色故障检测,导致其检测时延达到毫秒级.其次,在检测交换机外数据报丢失事件时,NS需要在交换机ASIC流水线中缓存已发送的数据报,并等待确认回复.由于交换机的资源限制,某些缓存的数据报在收到确认回前会被新到达的数据报覆盖,以致NS失效[2].

文献[20]提出了一种基于带内网络遥测(In-band Network Telemetry,INT)的网内灰色故障检测技术GrayINT(GI),实现了专用于数据中心Fat-Tree拓扑的灰色故障检测与定位.

GI首先基于INT获取Fat-Tree拓扑中的所有可用路径.其次,GI为每条路径设置一个计时器.当路径上没有数据报经过时,计时器的值会衰减.当某条路径p的计时器衰减到0时,GI怀疑p上发生了灰色故障.GI随后让p的一端主机发送INT探测报文.INT报文仅记录其经过的交换机标识符和出入端口编号,有助于减少INT引入的带宽开销.当超过一段时间阈值,p的另一端主机没有收到INT报文,GI则断定p上发生了灰色故障.此外,针对灰色故障的定位,GI将所有故障路径汇总至控制平面.通过识别所有故障路径中的共同节点,逐步将故障范围缩小至两个设备间的单一链路.

然而,GI存在着两个缺陷.首先,GI的灰色故障检测机制包含一轮计时器等待与一轮主动INT探测时延,导致其整体检测时延达到毫秒级.其次,虽然GI优化了其灰色故障定位算法,但仍需要控制平面介入灰色故障定位,导致其故障定位时延也达到毫秒级.此外,当收集到的故障路径较少时,GI无法有效地缩小灰色故障的定位范围.

前述技术主要针对低延迟网络(如数据中心),无法应对ISP网络中的大规模流量和链路时延.因此,文献[2]提出了一种用于检测ISP网络中灰色故障的技术FANcY(FC).FC可以在几秒钟内检测并定位大规模ISP网络中的灰色故障.

FC首先需要用户声明希望监控的条目以及交换机的资源上限.每个条目指定了一块数据报头部空间,表示用户希望检测属于此头部空间内的数据报是否丢失.由于每台交换机的资源有限,FC无法同时监控所有用户输入条目.因此,FC为每个条目制定了两种优先级:(1)最高优先级:FC会为这个等级的条目分配足

够的专用交换机计数器,能够准确地监控这些条目的数据报丢失事件;(2)尽力而为:在为最高优先级的条目分配计数器后,FC会使用剩下的资源为尽力而为等级的条目创建一棵公共的哈希树,监控这些条目的数据报丢失情况.两种优先级使得FC可以对所有网络流量实现全面覆盖.

在运行时,FC首先让上游交换机与下游交换机建立计数会话.在每个会话期间,上游交换机和下游交换机对每个条目的数据报进行独立计数.在每个会话结束时,下游交换机将其计数结果发送回上游交换机.上游交换机对比这些计数结果检测灰色故障,随后立即开启下一轮计数会话.

总结:本文从4个维度对现有灰色故障检测技术进行了分类:(1)检测时延,即检测到灰色故障所需要花费的时间;(2)检测精度,即检测灰色故障的准确性;(3)检测开销,即实现灰色故障检引入的额外开销(如带宽);(4)引入控制平面,即是否需要引入控制平面来辅助检测.分类结果如表2所示.结果表明:(1)与传统的、以控制平面为中心的灰色故障检测技术相比,基于可编程交换机的网内灰色故障检测技术能同时实现更短的检测时延、更高的精度和更低的开销;(2)完全依赖可编程交换机且不引入控制平面的灰色故障检测技术最具吸引力.

表2　不同灰色故障检测技术的对比

技术类型	技术方案	检测时延	检测精度	检测开销	引入控制平面
以控制平面为中心	镜像	高	高	高	是
	采样	高	低	低	是
基于可编程交换机	FlowRadar	高	高	低	是
	LossRadar	高	高	低	是
	Marple	高	高	高	是
	Blink	低	高	低	否
	NetSeer	高	高	低	是
	GrayINT	低	高	低	否
	FANcY	高	高	低	是

5　不同灰色故障技术的实验对比分析

本文通过一次真实的灰色故障检测案例对比不同灰色故障检测技术的性能,说明基于可编程交换机的网内灰色故障检测技术的优势.如图5所示,在数据平面,组合三台网络设备形成线性拓扑作为真实网络实验平台.其中,位于中央的设备是一台64×100 Gbps的Intel Barefoot Tofino可编程交换机.其余两台设备是运行着流量生成器和流量接收器的两台服务器.三台设备通过100 Gbps的链路直连.所有的服务器都配有128 GB内存和36核Intel Xeon中央处理器.

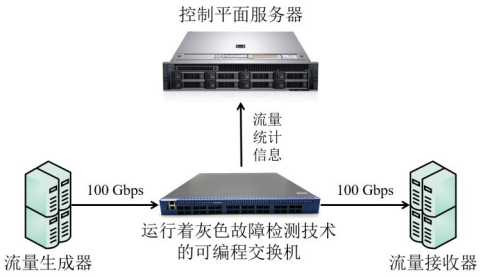

图5　真实网络实验平台

在实验过程中,流量生成器以100 Gbps的速率重放具有200万条流的CAIDA数据集.可编程交换机负责运行各种灰色故障检测技术.为了方便表示,本文将以控制平面为中心的两种灰色故障检测技术,即基于镜像的灰色故障检测技术和基于采样的灰色故障检测技术,分别用MR和SP表示.在控制平面,使用一台服务器从交换机收集数据并进行灰色故障检测.为了模拟灰色故障,配置交换机随机丢弃到达的数据报.在模拟灰色故障后,启动每种灰色故障检测技术进行故障检测.

由于某些检测技术(如MR和SP)需要协同来自多台交换机的统计数据来实现灰色故障的检测.本文借助可编程交换机的多流水线架构在一台设备上复现这些检测技术.具体而言,一台可编程交换机上配备四条数据报处理流水线,流水线之间支持互联且允许部署不同的数据报处理逻辑.因此,每条流水线可视作一台独立的交换机.如图6所示,本文在其中三条流水线上部署特定技术的检测模块,在一条流水线上部署随机丢包模块来模拟发生灰色故障的交换机,以此复现基于多台交换机协同的灰色故障检测技术.

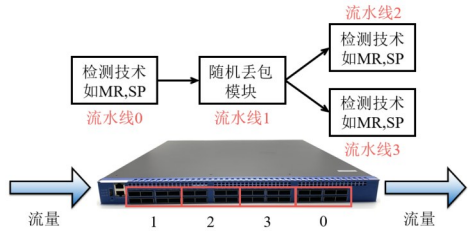

图6　基于多交换机协同灰色故障检测技术的复现方法

首先评估每种检测技术的检测时延.通过计算灰色故障发生时间与每种技术发现故障时间的差值,计算出每种技术的检测时延.图7展示了100次实验后每种技术检测时延的平均值.结果表明:(1)相比于以控制平面为中心的检测技术,基于可编程交换机的网内检测技术将检测时延缩短了几个数量级;(2)新技术的检测时延往往优于以往的技术;其中,BL和FC的检测时延是最短的,这得益于它们将整套灰色故障检测逻辑实现在可编程交换机上.

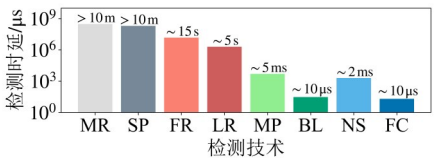

图7 灰色故障检测技术的检测时延

其次评估每种检测技术在检测过程中引入的额外带宽开销. 通过在控制平面服务器与流量接收器处截取流量,并将截取到的流量大小与数据集大小进行比较,计算出每种检测技术引入的额外带宽开销. 如图8所示,本文发现:基于可编程交换机的网内灰色故障检测技术在网内完成灰色故障的检测,不会引入额外的带宽开销. 而其他检测技术需要向控制平面汇报流量统计信息,引入了带宽开销. 其中,MR与MP的开销是最高的.

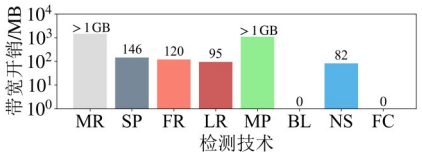

图8 灰色故障检测技术的额外带宽开销

最后评估每种检测技术的检测精度. 通过比较每种检测技术汇报的丢失数据报与实际丢失的数据报,计算每种检测技术的F_1分数. 图9表明:除了基于采样的灰色故障检测技术之外,其余技术的F_1分数均超过0.8,实现了高精度的灰色故障检测. 这是因为数据报采样牺牲了对流量的全局可见性,无法精确地捕捉到所有丢失的数据报.

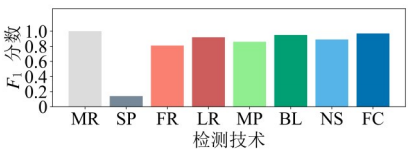

图9 灰色故障检测技术的检测精度

6 面临挑战

目前,基于可编程交换机的网内灰色故障检测技术已经取得了较多的研究成果,其应用也拓展到了数据中心网络和ISP网络中,这印证了此类新兴技术的可行性及有效性. 然而,基于可编程交换机的网内灰色故障检测技术的大规模应用和部署仍然面临着三个挑战.

(1)对异构网络的兼容性. 生产网络的数据平面普遍采用多样化的交换机配置. 这些配置不仅包括功能固定的传统交换机,还涉及到更为灵活的FPGA[30,31]. 特别是在数据中心网络环境中,这种多样化配置的普及性更为显著[32]. 然而,现有基于可编程交换机的网内灰色故障检测技术假设数据平面仅包含可编程交换机,忽略了可编程交换机以外设备的灰色故障检测机制的设计. 由于异构设备采用不同硬件架构,直接应用基于可编程交换机的故障检测技术是不现实的. 在只有部分节点可编程的异构网络中实现网内灰色故障检测技术,仍然是一个未解决且具有挑战性的问题.

(2)灰色故障修复. 本文介绍了基于可编程交换机的网内灰色故障检测技术. 这些技术主要关注灰色故障检测,如何修复这些故障仍是一个关键问题. 目前,这些技术在检测到灰色故障后,依赖于控制平面或者是用户指定的策略完成故障修复. 但由于控制平面和数据平面之间存在较高通信时延,导致修复策略无法及时地传达. 一个可能的解决方案是:利用可编程交换机的特性,在检测到灰色故障发生后,自动实施灰色故障修复策略. 可能的技术路线包括:(a)流量重定向:自动将流量重定向到健康路径,绕过故障点;(b)交换机备份:利用网络中备用可编程交换机,在主设备出现故障时自动接管流量,确保网络连续性.

(3)对其他类型灰色故障的支持. 本文重点关注导致交换机出现丢包的灰色故障及相关的网内检测技术. 然而,在生产网络中,还存在着其他类型的灰色故障,比如交换机处理时延增加、负载不均衡等[33]. 鉴于此,本文认为现有技术有必要支持检测除数据报丢失以外的其他类型的灰色故障. 具体而言,现有技术可以借助其网内检测的优势,对网络流量状态和交换机内部指标进行精细化捕捉,进而支持更多类型灰色故障的检测. 例如,可配置每台可编程交换机将处理每个数据报的时延附加到该报文的头部字段. 当发现某些流量的完成时间不符合预期时,分析其头部字段上所有交换机的附加的处理时延,及时发现性能瓶颈,或配置可编程交换机实时记录其发送端口的发送速率,当端口间的发送速率差异超过阈值时,及时汇报潜在的负载不均衡情况.

7 结束语

本文重点介绍了低时延、低开销和高精度的基于可编程交换机的网内灰色故障检测技术,详细阐述了基于可编程交换机的网内灰色故障检测技术的基本框架和工作流程,进一步分析了现有基于可编程交换机的网内灰色故障检测技术的最新研究进展及优缺点,通过实验评估验证了此类技术的优势,最后总结了此类技术面临的挑战.

参考文献

[1] 俞波, 杨珉, 王治, 等. 选择传递攻击中的异常丢包检测

[J].计算机学报, 2006, 29(9): 1542-1552.
YU B, YANG M, WANG Z, et al. Identify abnormal packet loss in selective forwarding attacks[J]. Chinese Journal of Computers, 2006, 29(9): 1542-1552. (in Chinese)

[2] MOLERO E C, VISSICCHIO S, VANBEVER L. FAst in-network gray failure detection for ISPs[C]//Proceedings of the ACM SIGCOMM 2022 Conference. New York: ACM. 2022, 677-692.

[3] HUANG P, GUO C X, ZHOU L D, et al. Gray failure: The Achilles' heel of cloud-scale systems[C]//Proceedings of the 16th Workshop on Hot Topics in Operating Systems. New York: ACM, 2017: 150-155.

[4] ARZANI B, CIRACI S, CHAMON L, et al. 007: Democratically finding the cause of packet drops[C]//15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18). Renton: USENIX Association, 2018: 419-435.

[5] ZHOU Y, SUN C, LIU H H, et al. Flow event telemetry on programmable data plane[C]//Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication. New York: ACM, 2020: 76-89.

[6] RASLEY J, STEPHENS B, DIXON C, et al. Planck: Millisecond-scale monitoring and control for commodity networks[J]. ACM SIGCOMM Computer Communication Review, 2014, 44(4): 407-418.

[7] ZHU Y B, KANG N X, CAO J X, et al. Packet-level telemetry in large datacenter networks[J]. ACM SIGCOMM Computer Communication Review, 2015, 45(4): 479-491.

[8] 林耘森箫, 毕军, 周禹, 等. 基于P4的可编程数据平面研究及其应用[J].计算机学报, 2019, 42(11): 2539-2560.
LIN Y, BI J, ZHOU Y, et al. Research and applications of programmable data plane based on P4[J]. Chinese Journal of Computers, 2019, 42(11): 2539-2560. (in Chinese)

[9] ESTAN C, KEYS K, MOORE D, et al. Building a better NetFlow[J]. ACM SIGCOMM Computer Communication Review, 2004, 34(4): 245-256.

[10] 杨宏宇, 王泽霖, 张良, 等. 面向物联网的多协议僵尸网络检测方法[J]. 电子学报, 2023, 51(5): 1198-1206.
YANG H Y, WANG Z L, ZHANG L, et al. A multi-protocol botnet detection method for IoT[J]. Acta Electronica Sinica, 2023, 51(5): 1198-1206. (in Chinese)

[11] 王鹃, 王江, 焦虹阳, 等. 一种基于OpenFlow的SDN访问控制策略实时冲突检测与解决方法[J].计算机学报, 2015, 38(4): 872-883.
WANG J, WANG J, JIAO H Y, et al. A method of openflow-based real-time conflict detection and resolution for SDN access control policies[J]. Chinese Journal of Computers, 2015, 38(4): 872-883. (in Chinese)

[12] CHEN X, HUANG Q, ZHANG D, et al. ApproSync: Approximate state synchronization for programmable networks[C]//2020 IEEE 28th International Conference on Network Protocols (ICNP). Piscataway: IEEE, 2020: 1-12.

[13] BOSSHART P, GIBB G, KIM H S, et al. Forwarding metamorphosis[J]. ACM SIGCOMM Computer Communication Review, 2013, 43(4): 99-110.

[14] BOSSHART P, DALY D, GIBB G, et al. P4: Programming protocol-independent packet processors[J]. ACM SIGCOMM Computer Communication Review, 2014, 44(3): 87-95.

[15] 叶进, 王建新. 异构网络中丢包识别研究综述[J]. 计算机科学, 2006, 33(12): 19-22, 33.
YE J, WANG J X. The research of loss differentiation algorithm in heterogeneous networks[J]. Computer Science, 2006, 33(12): 19-22, 33. (in Chinese)

[16] 张昕怡, 潘恒, 谢高岗. 可编程网络数据平面技术进展[J]. 电信科学, 2022, 38(6): 42-50.
ZHANG X Y, PAN H, XIE G G. Progress in programmable network data plane[J]. Telecommunications Science, 2022, 38(6): 42-50. (in Chinese)

[17] CHEN X, WU C M, LIU X, et al. Empowering network security with programmable switches: A comprehensive survey[J]. IEEE Communications Surveys & Tutorials, 2023, 25(3): 1653-1704.

[18] 魏祥麟, 陈鸣, 范建华, 等. 数据中心网络的体系结构?[J]. 软件学报, 2013, 24(2): 295-316.
WEI X L, CHEN M, FAN J H, et al. Architecture of the data center network[J]. Journal of Software, 2013, 24(2): 295-316. (in Chinese)

[19] 李阿妮, 张晓, 赵晓南, 等. 面向IaaS的云计算系统可用性评估[J].计算机科学, 2016, 43(10): 33-39.
LI A N, ZHANG X, ZHAO X N, et al. Cloud computing system availability evaluation for IaaS[J]. Computer Science, 2016, 43(10): 33-39. (in Chinese)

[20] ZHANG K, SU W, SHI H, et al. GrayINT—detection and localization of gray failures via hybrid in-band network telemetry[C]//2023 24th Asia-Pacific Network Operations and Management Symposium (APNOMS). Piscataway: IEEE, 2023: 405-408.

[21] LIU J, HALLAHAN W, SCHLESINGER C, et al. P4V:

Practical verification for programmable data planes[C]// Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication. New York: ACM, 2018: 490-503.

[22] LI Y L, MIAO R, KIM C, et al. LossRadar: Fast detection of lost packets in data center networks[C]//Proceedings of the 12th International on Conference on emerging Networking EXperiments and Technologies. New York: ACM, 2016: 481-495.

[23] 张朝昆, 崔勇, 唐骞祎, 等. 软件定义网络(SDN)研究进展[J]. 软件学报, 2015, 26(1): 62-81.
ZHANG C K, CUI Y, TANG H Y, et al. State-of-the-art survey on software-defined networking (SDN)[J]. Journal of Software, 2015, 26(1): 62-81. (in Chinese)

[24] LIU Z, NAMKUNG H, NIKOLAIDIS G, et al. Jaqen: A high-performance switch-native approach for detecting and mitigating volumetric DDoS attacks with programmable switches[C]//30th USENIX Security Symposium (USENIX Security 21). Berkley: USENIX Association, 2021: 3829-3846.

[25] 毕军.P4与可编程数据平面: 回顾与展望[J]. 中国计算机学会通讯, 2019, 15(3): 76-80.

[26] 尼克·麦克欧文, 金昶勳, 高荣新. 用P4对数据平面进行编程[J]. 中国计算机学会通讯, 2016, 12(7): 12-20.

[27] LI Y, MIAO R, KIM C, et al. FlowRadar: a better NetFlow for data centers[C]//13th USENIX symposium on networked systems design and implementation (NSDI 16). Berkeley: USENIX Association, 2016: 311-324.

[28] NARAYANA S, SIVARAMAN A, NATHAN V, et al. Language-directed hardware design for network performance monitoring[C]//Proceedings of the Conference of the ACM Special Interest Group on Data Communication. New York: ACM, 2017: 85-98.

[29] HOLTERBACH T, MOLERO E C, APOSTOLAKI M, et al. Blink: Fast connectivity recovery entirely in the data plane[C]//16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19). Boston: USENIX Association, 2019: 161-176.

[30] 田春生, 陈雷, 王源, 等. 面向FPGA的布局与布线技术研究综述[J]. 电子学报, 2022, 50(5): 1243-1254.
TIAN C S, CHEN L, WANG Y, et al. Review on Technology of Placement and Routing for the FPGA[J]. Acta Electronica Sinica, 2022, 50(5): 1243-1254. (in Chinese)

[31] 王鹏, 邹彬, 刘金枝, 等. 基于Xilinx型FPGA系统单粒子效应评估方法研究[J]. 电子学报, 2022, 50(11): 2716-2721.
WANG P, ZOU B, LIU J Z, et al. Study on single event effect evaluation method based on Xilinx FPGA system[J]. Acta Electronica Sinica, 2022, 50(11): 2716-2721. (in Chinese)

[32] AGARWAL A, LIU Z, SESHAN S. HeteroSketch: Coordinating network-wide monitoring in heterogeneous and dynamic networks[C]//19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22). Renton: USENIX Association, 2022: 719-741.

[33] 贾统, 李影, 吴中海. 基于日志数据的分布式软件系统故障诊断综述[J]. 软件学报, 2020, 31(7): 1997-2018.
JIA T, LI Y, WU Z H. Survey of state-of-the-art log-based failure diagnosis[J]. Journal of Software, 2020, 31 (7): 1997-2018. (in Chinese)

作者简介

刘宏岩 男, 1998年8月出生于辽宁省铁岭市. 现为浙江大学计算机科学与技术学院博士生. 主要研究方向为可编程网络、网络测量、网络安全.
E-mail: hyliu20@zju.edu.cn

张 栋 男, 1982年10月出生于福建省福州市. 现为福州大学计算机与大数据学院/软件学院教授, 博士生导师. 主要研究方向为软件定义网络、医学人工智能.
E-mail: zhangdong@fzu.edu.cn

吴春明 男, 1966年6月出生于浙江省杭州市. 现为浙江大学计算机科学与技术学院教授, 博士生导师. 主要研究方向为新一代互联网体系架构、可编程网络技术、网络系统内生安全.
E-mail: wuchunming@zju.edu.cn