计算机科学
COMPUTER SCIENCE

# 基于可编程数据平面的SRv6功能一致性验证机制

王鹏睿，胡宇翔，崔鹏帅，董永吉，夏计强

引用本文

王鹏睿，胡宇翔，崔鹏帅，董永吉，夏计强. 基于可编程数据平面的SRv6功能一致性验证机制[J]. 计算机科学，2025，52(10)：328-335.

WANG Pengrui, HU Yuxiang, CUI Pengshuai, DONG Yongji, XIA Jiqiang. SRv6 Functional Conformance Verification Mechanism Based on the Programmable Data Plane [J]. Computer Science, 2025, 52(10): 328-335.

---

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.240800163

# 基于可编程数据平面的 SRv6 功能一致性验证机制

王鹏睿[1] 胡宇翔[1,2,3] 崔鹏帅[1,2,3] 董永吉[1,2,3] 夏计强[1]
1 信息工程大学信息技术研究所 郑州 450002
2 先进通信网全国重点实验室 郑州 450002
3 网络空间安全教育部重点实验室 郑州 450002
(wprxxgcdx@163.com)

**摘 要** 现阶段 SRv6(Segment Routing over IPv6)中,段标签(Segment Identifier,SID)设计为流量工程、安全认证等网络功能提供了可编程性。这些功能的实现依赖于数据平面中流表的精确匹配与执行,但当流表被恶意篡改或错误配置时,容易导致功能一致性问题的出现。而带内网络遥测(In-band Network Telemetry,INT)技术作为 SDN 场景中经典的具有可编程性的校验工具,可将二者天然结合。为此,提出了基于可编程数据平面的 SRv6 功能一致性验证机制(Programmable Data Plane Based Consistency Verification Mechanism for SRv6 Functional,SRv6FCV)。SRv6FCV 采用数据平面可编程技术为探针包插入认证标识,首先依据监控需求动态地将 SID 转换为特定的 INT 元数据结构,然后构造探针报文注入并逐跳收集特定网络功能的流表执行信息,最后对遥测信息进行解析并基于符号执行算法完成功能一致性验证。仿真结果表明,SRv6FCV 能够保证满足流表规则以及业务功能执行策略的一致性。相较于之前的研究,SRv6FCV 在完成对网络功能一致性校验的基础上,拥有更低的运行开销,同时其校验时长也有显著缩短。

**关键词**:段路由;段标签;SRv6;带内网络遥测;一致性
**中图分类号** TP393

## SRv6 Functional Conformance Verification Mechanism Based on the Programmable Data Plane

WANG Pengrui[1],HU Yuxiang[1,2,3],CUI Pengshuai[1,2,3],DONG Yongji[1,2,3] and XIA Jiqiang[1]
1 Institute of Information Technology Research,Information Engineering University,Zhengzhou 450002,China
2 National Key Laboratory of Advanced Communication Networks,Zhengzhou 450002,China
3 Key Laboratory of Cyberspace Security,Ministry of Education,Zhengzhou 450002,China

**Abstract** At present,the SID in SRv6 is designed to provide programmability for traffic engineering,security authentication,and other network functions. The realization of these functions depends on the precise matching and execution of flow tables in the data plane,but when the flow tables are maliciously modified or incorrectly configured,it is easy to cause inconsistency problems in function implementation. As a classic verification tool with programmability in SDN scenarios,the INT technology can naturally combine with the two. This paper proposes the SRv6 Function Consistency Verification(SRv6FCV) mechanism based on programmable data plane. SRv6FCV uses data plane programmability technology to insert authentication identifiers into probe packets,first dynamically converts the SID into a specific INT metadata structure according to the monitoring needs,then constructs probe packets and injects them into the network to collect flow table execution information for specific network functions,and finally decodes the telemetry information and completes the function consistency verification based on symbolic execution algorithms. Simulation results show that SRv6FCV can ensure consistency between flow table rules and business function execution policies. Compared with previous studies,SRv6FCV,in addition to achieving consistency verification of network functions,has lower running overhead and significantly reduces verification time.

**Keywords** Segment routing,SID,SRv6,In-band network telemetry,Consistency

到稿日期:2024-08-30 返修日期:2024-11-25
基金项目:国家重点研发计划(2023YFB2903902);中原科技创新领军人才项目(244200510038);嵩山实验室重点研发项目(221100210900-02)
This work was supported by the National Key Research and Development Program of China(2023YFB2903902), Science and Technology Innovation Leading Talents Subsidy Project of Central Plains(244200510038) and Key R&D Projects of Songshan Laboratory(221100210900-02).
通信作者:胡宇翔(chxachxa@126.com)

## 1 引言

随着网络技术的快速发展，段路由（Segment Routing，SR）因其高效的网络路径控制能力和灵活性，在未来的网络架构中扮演着重要角色。段路由是一种实现 SDN 的源路由方法。在网段路由中[1]，一个网络被划分为多个网段，每个网段被分配一个唯一的 ID，称为 SID（Segment ID）。后续每个可编程数据平面（Programmable Data Plane，PDP）交换机根据带有 SID 的正确 SR 标签来转发数据包。SR 数据平面有两种实现方式：SR-MPLS（Multiprotocol Label Switching）和 SRv6。与 SR-MPLS 相比，SRv6 实现了带有标准 IPv6 报头的扩展，并使用 IPv6 地址作为 SID[2]。SRv6 作为一种基于 IPv6 和段路由的新一代 IP 承载协议，不仅简化了传统复杂的网络协议，还显著增强了网络的灵活性和可扩展性[3]。

SRv6 技术的核心在于利用 IPv6 报头的扩展功能，作为 SDN（软件定义网络）在 IPv6 转发平面的应用，SRv6 充分利用了 IPv6 的特性和 SDN 的设计理念，为网络带来了更高的灵活性、可扩展性和管理效率。在此引入 SRv6 功能的概念，即将 SID 嵌入 IPv6 报头中，使 SRv6 在 SRH（段路由头）中实现了自定义编程[3-4]，可以根据不同的业务需求提供灵活多样的服务，实现不同的功能，如路径选择功能即为实现对数据转发路径的精确控制。而流表作为 SDN 架构中数据平面的关键组成部分，且作为网络功能的实现载体，负责实现数据包的转发和处理，流表的精确匹配与执行对网络功能的正常运行至关重要。在数据平面，可编程数据平面的引入增加了 SDN 数据平面潜在的流表（功能）一致性风险。这些情况都会导致数据平面出现与控制平面既定策略（功能）不一致的转发行为，进而危及网络安全。

网络功能不一致的问题的主要原因是流表被（网络攻击）恶意篡改或错误配置，文献[5-6]从模型检查和符号执行的角度对 SDN 应用中存在的网络控制功能不一致性问题进行识别和分析。SDN 场景下，流表作为控制平面和数据平面之间的“桥梁”，针对流规则的一致性问题在（Programming Protocol-independent Packet Processors）P4 等数据平面编程语言出现之前已经得到广泛关注[7-12]。Perešíni 等[13]通过将交换机的转发表逻辑转换为命题可满足性问题（SAT）来创建专门的探针，用于验证目标流规则的正确性。这些探针不仅能够追踪流规则在更新过程中的加载状态，还能在网络安全稳定的条件下验证已加载流规则的有效性。其采用了一种主动探测的一致性检查机制，用于验证控制器已经同步的流规则。然而，这种机制无法检测并验证数据平面上因错误配置的流规则而导致的问题。

SDN 场景中，带内网络遥测（INT）技术被广泛应用于网络测量[14]。与传统技术相比，INT 将数据包转发与测量相结合，允许转发节点收集网络状态并在数据包中插入遥测数据，而且具有实时性强、测量粒度精细和状态丰富的优点[15]，这与 SRv6 中 SID 的功能结构具有明显的相似性。然而，INT 也面临数据包开销较大的问题。由于需要添加路径上每个交换机的信息，数据包的开销随路径的增加而增长，从而导致较高的带宽开销。例如，重复插入 INT 字段会使数据包的长度过长，在控制通道中产生较大带宽开销[16]。显然，这类方法在处理路径选择功能不一致问题时，带宽开销过高；文献[17]提出了一种在 SDN 场景下基于 P4 的流规则一致性校验机制，其进行校验的一致性算法依托于符号执行，能够完成对控制平面流规则配置和数据平面遥测信息的一致性校验。然而，该方法在计算效率方面仍有改进空间，并且不适用于 IPv6 场景。

SRv6 作为一种 IPv6 的原生应用，具有良好的兼容性，可以通过 128 位头部字段实现网络编程能力[18-19]。在此基础上，结合对 SRv6 等现代网络技术的研究，本文提出了一种基于可编程数据平面的 SRv6 功能一致性验证机制（Programmable Data Plane Based Consistency Verification Mechanism for SRv6 Functional SRv6FCV）。该机制通过定制 SRv6FCV 探针包头来满足各种遥测需求，从而实现低开销、灵活且准确的网络遥测，确保网络功能的一致性校验。与先前的针对校验流规则的一致性相比，SRv6FCV 针对 IPv6 场景进行了优化，降低了带宽消耗，缩短了校验时间，并提高了校验的准确性，同时实现了网络功能的一致性校验。此外，还使用 BMv2 软件交换机和 P4 语言进行了模拟仿真，并对实验结果进行了详细的分析。

本文的主要贡献可以概括为：

1）设计了一种适用于 SRv6 场景下的功能一致性校验方法，实现了对 SDN 可编程数据平面进行转发的功能一致性校验。

2）在 SRv6 场景下将转发路径的关键节点设置为探测转发路径的中间节点，具体实现验证路径。在此基础上，对 SRv6 数据包的字段进行修改，使其能以较低的开销携带遥测数据。

3）基于 Mininet 软件交换机搭建了仿真模拟系统。实验结果显示，与传统的网络测量方案相比，SRv6FCV 不仅能够实现核心业务功能，保证校验的准确率，而且在减小运行开销的同时有效缩短了校验时间，SRv6FCV 的校验时间相较于 SRCV 方法缩短了 27.27%，相较于 SR-INT 方法缩短了 17.11%。

## 2 研究背景

导致控制平面和数据平面出现网络功能不一致的原因主要包括以下几种情况。1）设备软硬件故障：交换设备在承载网络基础转发功能时，可能会遇到硬件故障，导致特定数据包被丢弃，形成转发黑洞，以及发生在设备操作系统中的错误。例如，当交换机的硬件转发表已满，新加载的流规则将被存入软件转发表，由于硬件表查询速度更快，交换机会优先检查硬件表。这样可能导致数据包匹配到硬件表中较低优先级的规则，而不是软件表中的较高优先级规则，从而影响转发效率和准确性[20]。2）网络攻击：数据平面是 SDN 架构中的基础功能平台，负责实际的数据包转发，几乎所有针对 SDN 的网络攻击最终都会影响到数据平面的正常运作，特别是一些常见的攻击手段会直接破坏数据平面流表的一致性和有效性，如洪泛攻击等。3）诸如控制平面东西向通信标准不统一而造成的控制器状态不一致等[21]。鉴于以上情况，本文遂开展了

基于可编程数据平面的 SRv6 功能一致性验证机制。

## 3 总体框架

在段路由场景下，流表内容的可定义性(如 SID)是实现灵活和高效路由的关键。然而，这种灵活性也带来了安全风险，如攻击者可以通过各种手段篡改交换机或路由器上的 SID 配置。SID 被篡改，原本规划的路径可能不再有效，可能导致数据包被错误地转发到非预期的目的地，使网络流量偏离预期路径，影响服务质量甚至造成安全漏洞，进而使其无法完成原有的路由功能，从而引发路由传输与计划不一致的问题。因此，在段路由场景下，对路由功能进行一致性校验是非常必要的。对此，本文提出了面向段路由场景的一致性校验方法 SRv6FCV。

SRv6FCV 部署流程如图 1 所示。

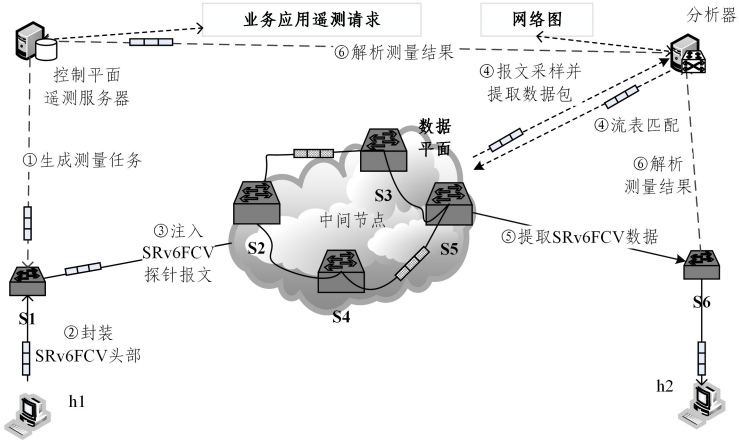


图 1 SRv6FCV 部署流程

Fig. 1 Deployment flow chart of SRv6FCV

当控制平面遥测服务器接收到业务应用发出的遥测请求后，会对该请求进行分析处理。随后，通过发送端的流量生成器产生主动探测流量，这种流量由五元组构成，目的是收集特定的目标和路径信息。根据收集的信息，这些五元组流会根据 SR 规则，从指定源点传输到目的节点。

在这个流程中，探针报文设计模块与信息收集模块合作，将 INT 头部和 SRv6 头部结合起来，生成 SRv6FCV 探针报文。这些报文随后被发送给 SRv6 列表中的首个交换机，每个途经的交换机都会在其报头中实时添加遥测数据。在数据平面，交换机首先解析探针报文，在报文中依次附加其自身的信息，然后根据 SRv6 列表进行转发，直到报文经过列表中的所有交换机。当流量到达目的地时，会对报文进行采样并提取包含路径信息和规则编码的数据包，然后将这些数据包转发给分析器以生成实际报告。

控制平面模块基于当前的拓扑结构和配置信息生成网络图，并发送给分析器。最后，探针由最后一个交换机送回至控制平面服务器。此时，信息收集模块解析探针，提取所有遥测元数据，并将其传递给校验模块。校验模块负责将这些元数据存入数据库，并将遥测报告反馈给上层应用。分析器根据实际报告构建出一个含有相似头部的符号包，并通过返回的路径模拟转发过程。这一过程能揭示各路径的状态，进而帮助分析器检测任何潜在的不一致性。

通过 SRv6FCV 方法，可以有效地简化一致性校验流程，实现更加细粒度的监控，提高网络灵活性以及可编程性，更好地利用 IPv6 的大地址空间并且简化数据包头部和配置，减小额外开销，实现故障检测与诊断。

## 4 模块设计

基于以上框架，将这个流程进一步细化成 3 个主要的功能模块：探针设计模块、信息收集模块以及功能校验模块。每个模块都有其特定的任务和职责，共同协作以确保 SRv6 网络的正确性和高效性。

1)探针设计模块

(1)功能设计：设计 SRv6 探针，用于在网络中收集遥测数据；SID 功能(FUN)字段：根据需要监控的网络行为，修改 SID(Segment ID)的 FUN 字段以指定探针的功能，例如可以设置 FUN 字段来指示探针应该收集哪些类型的遥测信息；编码：根据遥测内容和路径信息对探针的 INT 报头和 SRv6 报头进行编码，INT 报头包含网络路径中每跳的信息，SRv6 报头则定义了探针的路径。

(2)转发 SRv6 探针，用于执行必要数据包操作，探针根据其 SRv6 标签被转发到下一个节点；流量生成器：探针由流量生成器生成，并作为 SRv6FCV 的输入。

2)信息收集模块

(1)数据平面处理模块

数据包处理：接收探针，并执行必要的数据包处理逻辑，如解码 INT 报头中的遥测信息。转发：根据探针的 SRv6 报头中的路径信息，将探针转发到下一个节点。遥测数据集成：在探针中集成收集到的遥测数据，以便后续分析。

(2)控制平面处理模块

路径确定：依据给定的探针和源目标节点信息，确定探针的转发路径。策略制定：根据网络策略和业务需求，制定相应的转发策略。策略分发：将制定的策略分发到网络中的各个节点，以确保所有节点都能遵循统一的路由规则。

3)功能校验模块

(1)验证细节的读取

配置文件：从定义验证细节的文件中读取信息，这些文件包含了预期的网络行为和验证规则。一致性验证：验证节点

间通信的可达性以及路由功能的一致性。

(2)校验过程

路径可达性：检查路径上的每个节点是否可达，确保数据包能够沿着预期的路径顺利传输。转发行为一致性：确保所有节点遵循相同的路由策略，避免出现不一致的转发行为。

(3)结果输出

不一致信息：如果发现任何不一致的行为，例如数据包丢失、出现路径偏差等，输出所有相关的信息。确认信息：如果没有发现不一致的行为，则输出确认信息，表明数据平面上不存在不一致的转发行为。

通过上述 3 个模块的紧密合作，不仅可以实时监测网络状态，还可以有效地识别并解决潜在的问题，确保 SRv6 网络的稳定性和可靠性。这样的模块化设计有助于简化复杂的网络管理任务，并且便于未来的扩展和维护。

### 4.1 探针模块

在传统段路由网络环境中，在报文转发前会将路径每一跳地址写入头部 Segment List 字段中，再由指针指向下一跳地址，因此报文在网络传输中，每经过一跳指针便向下移动一次。但在该过程中，不难发现存在较为严重的资源浪费问题。例如该报文到达第 $n$ 个交换机时，Segment List[0]－Segment List[$n-1$]地址所代表的交换机已经完成转发任务，这些地址理论上已经失去价值，会造成 128 bit * $n$ 的资源浪费。

本项研究基于以上条件，对 SID 进行进一步改进设计，如表 1 所列。较为特殊的是 Segment List[0]，其中包含了一些功能字段信息，如路径选择功能字段、安全认证字段等，Flags 字段填充新的值，以辅助 SRv6FCV 的操作。由于目的是使 SRv6FCV 最小化每个数据包的长度，因此在数据包完成其路由路径上的段后，用 INT 元数据替换段列表。

表 1 SRv6FCV 报头格式

Table 1 SRv6FCV header format

| Eth | Ipv6 报头 | SRv6FCV | Payload |
|---|---|---|---|

| Next header | Hdr Ext Len | Routing type | Segment left |
|---|---|---|---|
| Last entry | Flags | Tag | |
| Segment list[0](IPv6 address,16bytes) | | | |
| Segment list[1](IPv6 address,16bytes) | | | |
| Segment list[2](IPv6 address,16bytes) | | | |
| … | | | |
| INTMetedata[0](Segment list[$n-1$]) | | | |
| Type | Length | Value | Determine (2bytes) | Reserved (2bytes) |

由表 1 可知，每个 Segment List 为 16 字节，存储了一个段的最后一个节点(即端点)的 IPv6 地址，Segment left(SL)表示数据包路由路径上的段总数。其他字段的定义如下。

Type：占用 8bit 的空间。该字段用于表示 INT 数据包类型。本文中为随流检测的基于 SRv6 扩展后的 INT 数据包。

Length：占用 8 bit 的空间。该字段用于表示 INT 头部长度，可根据该段判断 INT 数据从何处开始。

Value：该字段使用一个 1 字节的位图表示在每个段的最后一个交换机收集 INT 数据的类型，即运营商可以灵活地使用该字段自定义其网络监控方案。

Reserved：保留位，在此为填充作用，填充 SRH 以使其长度成为 8 字节的倍数。

Determine：判定位，占用 2 字节，在此判定流表传输的功能一致性。第一位表示分类器是否预测正确，第二位表示分类器的预测结果。

INT Metadata：该字段大小为 8 字节，包含设备 ID、输出端口、Hop Latency(跳延迟)和带宽的 INT 数据。其可以实现路径追踪、性能监控、状态报告、流量分析等功能。

Segment List：与 INT Metadata 的长度相同，表示一个段的状态或信息包括设备 ID、时间戳、端口号、队列信息、性能统计数据。它由两个子字段组成，即 TTL 和 Label。TTL 使用一个字节来标识数据包在当前段中的位置。

为了支持 SRv6FCV，遵循 SRv6 的原则定义了 3 个动作，即 H. Encaps. INT，End. T. INT 和 End. DT. INT，用于修改 SRv6 域的入口和端点交换机上的数据包。算法 1 给出了 SRv6FCV 对报文的入口、端点和传输交换机的操作。第 3－7 行是 H. Encaps. INT 在入口交换机上的动作，它封装了 SRH 并相应地更新了外部 IPv6 报头。第 8－14 行描述了 End. T. INT 对端点交换机的操作。第 15－20 行用于 End. DT. INT 在数据包路由路径上的最后一个端点交换机(即出口交换机)上的动作。

交换机接收到探测报文后，依次解析以太网报头、Ipv6 报头和 SRv6FCV 报头，根据指令位图字段在 SRv6FCV 报头后面附加相应的元数据，查询扩展报头的 Next header 字段，判断是否存在 SRv6 报头。如果存在，则将目的 IP 地址替换为 Segment left 所表示的 Segment list[$n$]，然后将 Segment Left 的值减 1，最终根据目的 IP 地址转发探测；如果不存在 SRv6 头，探针将直接根据原始 IPv6 目的地址转发。探针设计系统读取 SRv6 标签栈，逐个填充到 Segment List[$n$]中，并将 Hdr Ext Len 字段分配给 SRv6 标签栈的长度，实现 SRv6 报头设计。

**算法 1** Operations of SRv6FCV on a Switch

```
1. 接收数据包
2. If SRv6FCV is enabled then
3. If the packet's IPv6 header
       does not contain an SRH then
4.       // ingress switch:
5.       H. Encaps. INT encode SRH in
         IPv6 header and update
         outer IPv6 header;
6.       setSRH. Flags set SRH. Tag
7.     Else if SRH. Segment Left > 0
8.       then
9.       i=SRH. Segment Left;
10.      // endpoint switch:
11.      End. T. INT
12.      // plain SRv6FCV
13.      replace Segment List[i]
         With INTMetadata[SL-1-i];
14.      decreaseSRH. Segment
```

```
        Left by 1;
15.     update outer IPv6 header;
16.   Else
17.     // endpoint switch;
18.     End. DT. INT copy outer IPv6
        header with SRH to data analyzer;
19.     remove outer IPv6 header;
20.   End
21.     submit the packet to IPv6
22.     FIB lookup;
23. End
```

对算法的复杂度进行分析，整个算法的主要操作集中在条件判断、头部更新和 FIB 查找上。其中，处理 SRH. Segment Left 的部分可能会导致最多 $k$ 次迭代，每次迭代都是 O(1)操作。因此，该部分的复杂度为 O($k$)，其他所有操作均为 O(1)。综上所述，整体的时间复杂度为 O($k+1$)=O($k$)。其中 $k$ 代表 SRH. Segment Left 的最大值，即段路由路径中段的数量。

### 4.2 信息收集模块

#### 4.2.1 探针解析

当服务器接收到来自业务应用程序的遥测请求时，信息收集模块对遥测请求进行解析和处理，以获得遥测内容和链路信息。应用程序生成测量需求，包括目标网络设备和传输链路的列表，并将这些需求传递给控制平面以执行测量任务。最后，接收控制器的测量结果，进行进一步分析。本文通过在段路由(SRv6)场景下设置特定数量和结构的探针，来收集正常数据分组转发过程中匹配的流规则 ID、出入端口号等执行信息。这样做既简化了探针的结构，降低了数据平面的开销，又实现了对所需功能一致性校验信息的有效收集。

#### 4.2.2 探针注入和收集

当应用收到遥测请求时，它首先确定最优的遥测路径，并生成对应的 SIDs 和 Segment List。随后，应用根据遥测需求及路径信息，对 SRv6FCV 报头中的 SID 进行编码，并通过 SRv6 标签将该报头转发至下一跳交换机，构建探测报文，并发送给 SRv6 列表指定的第一个交换机。

在数据平面上，交换机会解析这些探针，按遥测指令的顺序将本地元数据添加到探针中，并依据 SRv6 标签列表进行转发，直至探针经过所有指定的交换机。信息收集模块会基于生成的段列表创建探测包，并将请求的范围和类型记录在表中。接着，利用路径规划算法计算 SRv6 标签堆栈，并查询每个节点的 IP 地址，将这些信息存入路径记录表中。之后，通过 OpenFlow 协议将这些配置信息下发给数据平面。最终，测量监控模块作为 INT 服务器接收从数据平面传来的 INT 遥测信息，并对其进行预处理后反馈给相关的应用程序。数据平面包括支持 SRv6 的可编程网络交换机，以支持 SRv6 和 INT 功能。网络设备包括接收控制平面探测报文的入口节点和提取 INT 元数据发送回上层的出口节点。更具体地说，目标网络设备应该同时支持 SRv6 和 INT。由于 IPv6 的兼容性，中间转发设备可以是普通的网络转发设备。

本文通过实施模块化的架构设计，实现了控制平面与数据平面之间的解耦，这是 SDN 的核心理念之一。这一设计使得控制平面能够以一种通用的方式管理数据平面上各网络设备的规则配置，而不依赖于具体的硬件特性。这意味着即便控制平面采用了不同格式的文件来保存转发规则，只需要调整控制平面模块内的解析逻辑即可确保系统的兼容性和灵活性。

### 4.3 功能校验模块

为了确保 SRv6 网络中的数据转发行为与预期一致，功能校验模块扮演着至关重要的角色。它通过对数据平面中的探针遥测数据与控制平面的配置信息进行比对，来验证数据平面内的转发功能一致性。此模块包含两个主要组件：路由功能一致性分析和符号执行。路由功能一致性分析主要是比对数据平面与控制平面信息，验证数据平面中的实际转发行为是否与控制平面中配置的期望行为相符。收集探针在网络中经过的每一条路径上的遥测数据，包括路径信息、中间节点的行为等，同时获取定义了预期的转发规则和策略的控制平面中的配置信息。而后逐节点检查，对于每个节点，检查其实际行为是否符合控制平面中配置的预期行为，标识出功能表现与预期不符的设备，并记录下来，最后进行校验结果汇总。符号执行则包含符号包创建，根据探针包的五元组信息(源 IP 地址、目的 IP 地址、源端口、目的端口、协议类型)创建一个具有相同头部字段的符号包(Symbolic Packet，SP)，SP 保留了探针包的关键属性，但并不携带实际的数据负载，而是用于模拟探针在网络中的传输过程；模拟传输过程则是沿着探针的实际转发路径模拟 SP 的传输过程，分析 SP 在每一步中的行为，以验证其是否按照预期的规则进行转发。当发现功能不一致的情况时，对整个路径下的所有不一致的功能节点信息进行输出。需提供关于不一致节点的具体信息，如节点的标识、不一致的位置和类型，帮助网络管理员快速定位问题。

## 5 实验设计

### 5.1 实验设置

实验采用了 BMv2 软件交换机构建网络拓扑，并在虚拟机中的 Mininet 仿真器上进行了测试。虚拟机配置如表 2 所列。

表 2 虚拟机配置

Table 2 Virtual machine configuration

| 设备 | 配置 |
|---|---|
| 操作系统 | Ubuntu 18.04 LTS |
| 处理器 | Intel(R) Core(TM) i7-10700 CPU @ 2.90GHz |
| 内存 | 8GB |

为了验证本文提出的基于可编程数据平面的 SRv6 功能一致性验证机制的可执行和高效率的特点，设置了两组实验，均包含 3 种方案：方案 1 是基于源路由的 SDN 运行一致性验证机制(SRCV)，方案 2 是基于段路由的网络测量遥测机制(SR-INT)，方案 3 为本文的 SRv6FCV 方案。第一组实验旨在对比 3 种方案的控制器开销；第二组实验评估了本文方案在 SRv6 场景下的一致性校验性能。这两组实验均采用了目前数据中心广泛使用的叶脊(Spine-Leaf)拓扑结构(见图 3)。实验控制校验路径节点数量为 5，首先由源节点将给定路径对应的源路由端口序列压入探针头部(其格式如表 1 所列)，然后周期性地发送一组探针，目的节点收到探针后进行采样、

解析并完成一致性校验。每组实验重复 20 次，记录从探针发出到 SRv6FCV 完成一致性校验的时间的校验总时长，将本文方案与 SRCV 机制和 SR-INT 机制进行对比，取实验的算数平均值作为最终实验结果。

由于流量工程、安全认证等路由传输的基础功能需要保证路由传输的安全准确可靠，因此将路由功能一致性指标量化为准确率(Accuracy，A)、查准率(Precision，P)以及查全率(Recall，R)。所有可能结果如表 3 所列。

表 3 判定位所有可能结果

Table 3 Identify all possible outcomes

| 结果 | Positive | Negative |
|---|---|---|
| True | True Positive(TP)<br>预测结果为正样本，实际也为正样本，即正样本被正确识别的数量 | TrueNegative(TN)<br>分类器预测结果为负样本，实际为负样本，即负样本被正确识别的数量 |
| False | FalsePositive(FP)<br>分类器预测结果为正样本，实际为负样本，即误报的负样本数 | FalseNegative(FN)<br>分类器预测结果为负样本，实际为正样本，即漏报的正样本数 |

将 Accuracy 记为预测正确的样本比例。

$$Accuracy=\frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

Precision 记为预测正确的正样本的准确度，查准率等于预测正确的正样本数量/所有预测为正样本的数量。

$$Precision=\frac{TP}{TP+FP} \tag{2}$$

Recall 记为预测正确的正样本的覆盖率，查全率等于预测正确的正样本数量/所有正样本的总和。

$$Recall=\frac{TP}{TP+FN} \tag{3}$$

本文同时考虑精确率(Precision)和查全率(Recall)两个指标，引入 F1 分数，通过更加平衡的视角来评估模型的性能，其计算式如下：

$$F1=2\times\frac{Precision*Recall}{Precision+Recall} \tag{4}$$

为了确保校验结果准确无误，SRv6FCV 对功能信息的校验涵盖了目的 IP 地址、出/入端口号以及流规则 ID 等关键要素。当功能信息匹配时，交换节点的校验被视为成功，并会更新状态信息(SP)中的相关字段，随后继续对下一个节点进行校验；若功能信息不匹配，则认为该节点的校验失败，并记录相应的故障信息，然后继续检查路径上的后续节点，以确定是否存在其他功能不一致的问题。

在实验中，设置网络转发节点数量为 5，网络流转发规则数量从 0 到 100000 动态变化，并均匀选取了检测值(20000，40000，60000，80000，100000)。为了确保校验结果的实时反馈，并同时降低探针数据包的发送延迟，源节点的探针发送速率被设定为每秒 100 个包(100 pps)。此外，实验中错误配置的注入是通过可编程交换机的命令行接口来手动修改其中的流规则，这些错误配置随机分布在转发路径的不同节点上。

在一致性校验过程中，系统会逐个对比探针经过的每一台交换机，并标记出流规则不一致的交换机。当一致性校验模块完成校验后，将输出每条转发路径的校验结果。如果某条路径上存在不一致的情况，系统不仅会标记第一个检测到的故障节点，还会继续检查路径下游的其他交换机，并最终输出该路径上所有故障节点的详细信息，包括节点 ID 和流规则信息等。

### 5.2 开销检测

结合段路由数据包转发前封装好路径这一特性，使用 SRv6FCV 可以轻松避开对可用带宽影响较大的“瓶颈链路”，从而得到较为理想的测量结果。SRv6FCV 作为监控网络运行状态的测量系统，在运行期间既不能占用过多的网络资源，也不能对正常业务流产生影响。因此本工作需要对 SRv6FCV 各方面的开销进行测试。本次实验首先评估了 SRv6FCV 在不同交换机数量下的控制器开销，以验证控制器是否会成为瓶颈。对于内存占用而言，如图 2 所示，控制器的内存占用随着交换机数量的增长呈线性增长，即使在 100 台交换机组成的网络中，SRv6FCV 的控制器内存占用仍少于 50 MB，控制器的内存占用对于总内存来说可以忽略不计。

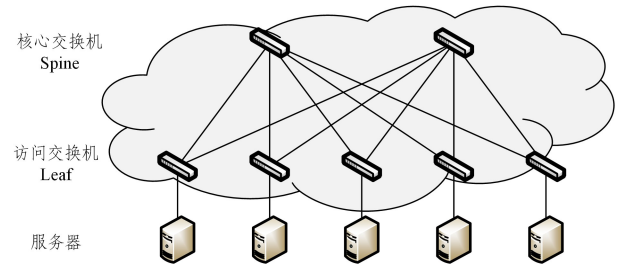


图 2 叶脊(Spine-Leaf)拓扑图

Fig. 2 Spine-Leaf topology map

从图 3 和图 4 的实验结果可以看出，符号执行校验的开销与该路径上交换节点的数量呈线性关系。由于符号执行采用顺序查表的方式，并且每台交换机内部配置了相同数量的流规则，在 CPU 占用率方面，如图 5 所示，当拓扑中交换机数目增加时，CPU 占用率显著上升，当交换机数目达到 100 时，在 SRv6FCV 机制下 CPU 占用率达到了 21.33%，显著低于另外两种对比方法。

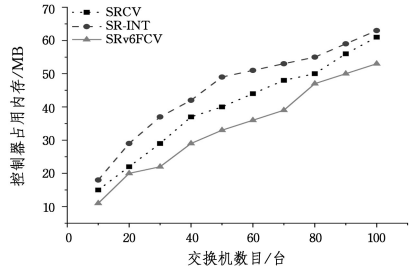


图 3 控制器内存占用

Fig. 3 Controller memory usage

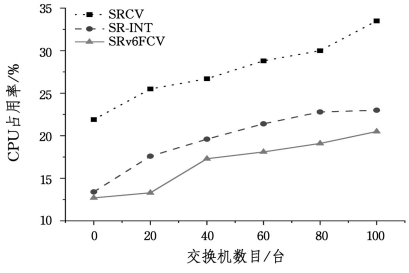


图 4 CPU 占用率

Fig. 4 CPU usage

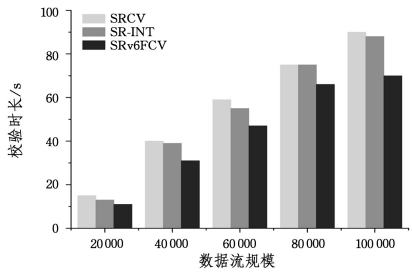


图 5 校验时长

Fig. 5 Verification duration

由于符号执行采用顺序查表的方式，且每台交换机中配置了相同数量的交换机。在性能测试中，本文关注的是随着数据流规模的增加，校验所需时间的变化。实验结果显示，当数据包数量线性增加时，校验时长也随之线性增长。然而，与两种对比方案相比，本文方案在校验时长上表现更为优越，即使在大规模数据流的情况下，其增长速度也明显低于对比方案。这表明本文方案在处理大量数据包时具有更好的扩展性和效率。由图 6 可以看出，SRv6FCV 与 SRCV 校验机制的校验时长均与校验流规则数量呈线性相关，随着数据流规模的增大，校验时长也在线性变化，且 SRv6FCV 的校验时间相较于 SRCV 方法缩短了 27.27%，相较于 SR-INT 方法缩短了 17.11%。

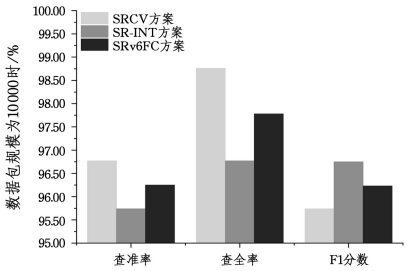


图 6 数据包规模为 10000 时的传输准确率

Fig. 6 Transmission accuracy when packet size is 10000

### 5.3 功能校验

由于数据流中，正负指标的数量差距较大，且传输准确率是衡量数据传输性能的关键指标，因此引入机器学习中常用的评估指标查准率 $P$、查全率 $R$、$F1$ 分数以及假阳性率辅助进行性能测试，其测量结果如图 7—图 8 所示。

在保持错误配置的注入不变的情况下，对于在同一时间段内注入的探针校验流，基于可编程数据平面的 SRv6 功能一致性验证机制与另外两种方法相比，在不同规模的数据集上所得到的误报率没有显著差异，并且该机制的精确度和召回率都能稳定维持在 95%以上。因此，总结得出，在功能校验实验中，我们评估了数据包规模对传输准确率的影响，包括查准率、查全率以及 F1 分数。实验结果表明，尽管数据包规模逐渐增大，但传输准确率依然能够保持在一个较高的水平。具体来说，无论是在小规模还是大规模的数据流中，本文方案都能够确保查准率、查全率及 F1 分数维持在高水平，从而证明了该方案在不同数据流规模下均能有效完成功能一致性校验。而不同数据包规模可实现不同路由功能，因此该方法在 SRv6 场景下，可以实现不同数据包规模的流规则一致性校验，即实现功能一致性校验。在达到相近准确性的前提下，SRv6FCV 机制的资源消耗较低，并且校验所需时间较短。

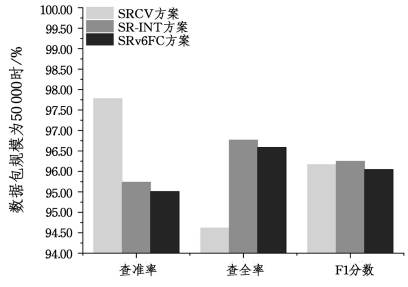


图 7 数据包规模为 50000 时的传输准确率

Fig. 7 Transmission accuracy when packet size is 50000

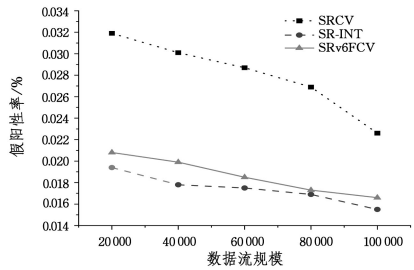


图 8 数据包规模变化的假阳性率

Fig. 8 False positive rate of packet size changes

**结束语** 本文针对以流表为载体产生的功能一致性问题，提出了一种基于可编程数据平面的 SRv6 功能一致性校验机制。该方案结合了带内遥测(INT)和 SRv6 报头来定制 SRv6 功能一致性校验 SRv6FCV 探针，灵活生成相应的遥测指令，指导不同遥测内容类型 SRv6FCV 头的设计，实现遥测内容的编排。通过这种方式，可以灵活地控制遥测内容的类型和格式，实现对遥测内容的精准编排，避免了不必要的数据采集，更精确地监控网络行为。最后，通过仿真验证了该方案相较于传统 INT 方案能够在有效缩短校验时长的同时，不会对核心业务产生较大影响，减少运行开销，且能够保证 SRv6FCV 的校验准确率。

虽然本文提出的 SRv6FCV 机制在流规则一致性校验方面表现出色，但仍存在一些局限性：开销较大、流内信息冗余以及流间路径重复。为了解决这些问题，接下来的研究可以利用段路由技术(例如 SRv6)的转发特性，通过将预先规划好的路径直接编码到 SID 中，确保数据包按照指定路径进行传输。这种方法可以有效减少不必要的路径重复和信息冗余，从而降低整体开销。

除此之外，实验的安全性也是下一步研究的重点，如段路由将路径信息明文写在报文头部，其可能造成路径信息泄露、篡改路径信息、欺骗源地址、拒绝服务攻击等风险。接下来的工作，可以利用 SID 的可编程性，完成对路径信息的加密，或者完成访问控制机制以及多路径选择与故障恢复，以有效降低源路由方式带来的安全风险，保障网络通信的安全性和可靠性。

对于未来研究，有以下几个方向值得深入探讨：研究如何在 SRv6 网络中实现路径的一致性校验，确保数据包按照预期路径转发；安全策略一致性校验，探讨如何检测和防止安全

策略被篡改或错误配置，保障网络安全；或者是面向特定场景的一致性校验，即针对边缘计算、算力网络等特定场景，开发适应性强的一致性校验机制。

## 参 考 文 献


[1] SUGIURAT，TAKAHASHI K，ICHIKAWA K，et al. Acar：An application-aware network routing system using SRv6 [C] // 2022 IEEE 19th Annual Consumer Communications & Networking Conference(CCNC). 2022：751-752.

[2] ZHENGQ，TANG S，CHEN B，et al. Highly-Efficient and Adaptive Network Monitoring：When INT Meets Segment Routing [J]. IEEE Transactions on Network and Service Management，2021，18(3)：2587-2597.

[3] WEI W，ZHANG X，PAN P，et al. EPM-SR：efficient performance measurement framework for KPIs to support segment routing over IPv6 Network[C] // 2022 IEEE 22nd International Conference on Communication Technology(ICCT). 2022：1800-1805.

[4] CHEN B，CHEN F，TANG S，et al. On Orchestration of Segment Routing and In-band Network Telemetry[J]. IEEE Transactions on Network and Service Management，2023，20(4)：4047-4060.

[5] ZUO Q Y，CHEN M，ZHAO G S，et al. Openflow-based SDN technologies[J]. Journal of Software，2013，24(5)：1078-1097.

[6] BIFULCO R，RÉTVÁRI G. A survey on the programmable data plane：Abstractions，architectures，and open problems[C] // Proceedings of the 19th IEEE International Conference on High Performance Switching and Routing(HPSR). IEEE，2018. 1-7.

[7] WANG X Y，HU A Q，FANG H. Improved collusion-resistant unidirectional proxy re-encryption scheme from lattice[J]. IET Information Security，2020，18(1)：342-351.

[8] DUTTA P，SUSILO W，DUONG D H，et al. Collusion-resistant identitybased proxy re-encryption：lattice-based constructions in standard model[J]. Theoretical Computer Science，2021，871：16-29.

[9] WANG X A，GE Y L，YANG X Y. PRE +：dual of proxy re-encryption and its application[J]. International Journal of Web and Grid Services，2018，14(1)：44-69.

[10] SINGH K，RANGAN C P，AGRAWAL R，et al. Provably secure lattice based identity based unidirectional PRE and PRE + schemes[J]. Journal of Information Security and Applications，2020，54(3/4)：102569.

[11] ATENIESE G，FU K，GREEN M，et al. Improved proxy re-encryption schemes with applications to secure distributed storage [J]. ACM Trans on Information and System Security，2006，9(1)：1-30.

[12] GUO H，ZHANG Z F，XU J，et al. Non-transferable proxy reencryption[J]. The Computer Journal，2019，62(4)：490-506.

[13] PEREŠÍNI P，KUŹNIAR M，AND KOSTIĆ D. Monocle：Dynamic，Fine-grained Data Plane Monitoring[C] // Proceedings of CoNEXT. 2015：1-13.

[14] TAN L，SU W，MIAO J，et al. FindINT：Detect and Locate the Lost in-Band Network Telemetry Packet[J]. IEEE Networking Letters，2022，4(1)：20-24.

[15] MARQUES JA，GASPARY L P. Advancing Network Monitoring and Operation with In-band Network Telemetry and Data Plane Programmability[C] // NOMS 2023-2023 IEEE/IFIP Network Operations and Management Symposium. 2023：1-6.

[16] LIU W，ZHANG X，FENG C，et al. Segment Routing based In-Band Network Telemetry in IPv6 over Optical Networks[C] // 2024 2nd International Conference On Mobile Internet，Cloud Computing and Information Security(MICCIS). 2024：125-129.

[17] GENTRY C，PEIKERT C，VAIKUNTANATHAN V. How to use a short basis：trapdoors for hard lattices and new cryptographic constructions[C] // Proc. of the 40th ACM Symposium on Theory of Computing. 2018：197-206.

[18] WANG F H，HU Y P，JIA Y Y. Lattice-based signature scheme in the standard model[J]. Journal of Xidian University，2012，39(4)：57-61，119.

[19] QIU L S，WANG L L，LIU J，et al. SRSV：Efficient Resource Reservation for Satellite Networks Based on Segment Routing [C] // 2022 5th International Conference on Hot Information-Centric Networking(HotICN). 2022：99-104.

[20] WANG X Y，HU A Q，HAO F. Feasibility analysis of lattice-based proxy re-encryption[C] // Proc. of the 17th International Conference on Cryptography，Security and Privacy. 2017：12-16.

[21] CHICA J C C，IMBACHI J C，VEGA J F B. Security in SDN：A comprehensive survey[J]. Journal of Network and Computer Applications，2020，159：102595.


**WANG Pengrui**，born in 1997，postgraduate. His main research interests include segment routing and programmable data plane.

**HU Yuxiang**，born in 1982，Ph.D，Ph.D supervisor. His main research interests include next generation network architecture and switching technology.

（责任编辑：喻藜）