PIFET-协议无关的灵活加密传输机制

刘泽英,崔鹏帅,胡宇翔,董永吉,王 钰,李子勇
(中国人民解放军战略支援部队信息工程大学 信息技术研究所,河南 郑州 450000)

摘 要:随着网络技术的发展,各类新型网络协议层出不穷。然而,当前异构网络缺乏完善、有效的安全机制,数据传输过程中面临着隐私泄露的风险,可能带来一系列安全问题。针对上述挑战,该文提出了一种协议无关的灵活加密传输机制(PIFET),通过提高数据的机密性和加密的灵活性,进一步保障异构数据的通信安全。首先,基于可编程平台设计了面向异构协议的灵活加密传输的系统架构;在此基础之上,根据异构数据的传输需要和安全需求,实现了灵活的加密机制,提供了两种不同安全等级的加密方法;最后,提出了面向隧道模式的字段灵活可选的加密机制。实验结果表明,PIFET 为用户提供了多种安全级别的、可定义的加密方法,满足了不同数据类型的加密需求。字段灵活可选的加密机制减少了不必要的加密量,从而降低了延迟,提高了系统的时间效率。

关键词:SDN;可编程数据平面;灵活加密;异构协议;安全策略
中图分类号:TP393 **文献标识码**:A **文章编号**:1673-629X(2023)11-0106-07
doi:10.3969/j.issn.1673-629X.2023.11.016

PIFET-Protocol Independent Flexible Encrypted Transmission Mechanism

LIU Ze-ying,CUI Peng-shuai,HU Yu-xiang,DONG Yong-ji,WANG Yu,LI Zi-yong
(Institute of Information Technology Research,People's Liberation Army Strategic Support Force Information Engineering University,Zhengzhou 450000,China)

Abstract:With the development of network technology,various new network protocols have emerged. However,current heterogeneous networks lack perfect and effective security mechanisms,and face the risk of privacy leakage during data transmission. To address the above challenges,we propose a protocol-independent flexible encrypted transmission mechanism (PIFET) to further secure the communication of heterogeneous data by improving the confidentiality of data and the flexibility of encryption. Firstly,we design a system architecture for flexible encrypted transmission of heterogeneous protocols based on a programmable platform. Besides,a flexible encryption mechanism is implemented according to the transmission needs and security requirements of heterogeneous data,and two encryption methods with different security levels are provided. Finally,a fields-flexible optional encryption mechanism oriented to tunnel mode is proposed. The experimental results show that PIFET provides users with multiple security levels and definable encryption methods to meet the encryption needs of different data types. The fields-flexible optional encryption mechanism of the fields reduces the amount of unnecessary encryption thus reducing the latency and improving the time efficiency of the system.

Key words:SDN;programmable data plane;flexible encryption;heterogeneous protocols;security policy

0 引 言

随着网络技术与经济社会的不断发展,特别是“工业 4.0”,“互联网+”与“网络 5.0”的深度融合,现有的网络结构及由此构建的技术体系存在“IP 单一承载”“身份与位置绑定”等诸多原始设计缺陷,难以满足多元化的网络应用场景[1]。为了从体系上克服上述问题,学者对新型网络体制进行了大量的研究。1990 年,麻省理工大学 Tennenhouse 提出了 ALF(Application Layer Framing),该架构将表示层定义为整个协议性能的关键层次,并赋予 OSI 模型新的定义。接着,主动网络,Data-Oriented Network Architecture (DONA),Named Data Networking (NDN)[2],MobilityFirst (MF)[3],Scalability,Control,and Isolation (SCION)[4],标识网络,GeoNetworking(Geo)[5] 等多

收稿日期:2022-11-20 **修回日期**:2023-03-22
基金项目:国家重点研发计划(2022YFB2901403);嵩山实验室项目(221100210900-02)
作者简介:刘泽英(1996-),男,硕士研究生,通讯作者,研究方向为新型网络体系结构、路由与交换技术;崔鹏帅,博士,副研究员,研究方向为新型网络体系结构、路由与交换技术;胡宇翔,博士,教授,研究方向为新型网络体系结构、路由与交换技术。

种网络体制持续涌现,分别从网络的可扩展性、高效性、移动性和互联性等方面提出了新的思路。其中 Geo 和 NDN 已取得了小规模的部署。新型网络体制的出现和应用奠定了多体制网络并存的技术基础,不同网络体制共存的异构网络也将长期存在。

然而,多种新型网络体制不断涌现的同时网络也面临着更加严峻的安全隐患[6]。MF 仅对 GUID 身份的唯一性进行验证,缺乏身份、流量等关键数据的分析与处理能力,在高复杂度和高动态的网络中对信息窃取和篡改等网络攻击方式缺乏有效的防御手段[7]。Geo 在实际应用过程中,其安全机制仅对发送消息做数字签名来验证发送方的身份,而数据包仍然以明文形式传输,在公共网络中面临着被窃取的风险[8]。另一方面,当前互联网缺乏不同网络协议通用的安全传输机制。IPsec 虽然能够保障网络层数据的安全性,但只针对 IP 协议有效,并不适用其他网络体制。所以,新型网络协议数据在传输过程中面临着隐私泄露的风险。

加密技术是信息安全技术的核心,安全传输技术通过应用不同的加密协议和机制来保护信息的机密性和完整性。当前广泛使用的加密机制主要有两种。一是采用端点加密的方式加密所有数据包负载。然而,考虑到终端用户、网络流量等因素差异,并非所有数据流都需要安全保护,该方法无法根据数据包的重要性来决定是否添加安全保护。此外,基于负载的加密机制虽能保护有效载荷,但由于包头部分未加密,攻击者依然能够利用网络探测等技术获得路由信息。第二种是数据的选择性加密,即根据数据包不同字段所携带信息的重要性,对包头在内的所有字段选择性加密。然而,当前的选择性加密方法主要应用于单一、特定的网络体制,不同网络体制的数据包结构千差万别,并不具有通用性。所以,目前仍然缺乏面向异构网络协议的通用、灵活的安全传输机制。

为解决该问题,该文提出了协议无关的灵活传输机制(Protocol Independent Flexible Encryption Transmission Mechanism,PIFET),并成功在可编程交换平台上实现了原型系统的开发和评估。主要贡献如下:

(1)设计了基于 P4 可编程平台的整体系统架构,包括数据平面的完整处理逻辑和控制器的模块化功能设计;(2)实现了面向异构协议的灵活加密机制,用户可以根据加密需求自定义加密策略和加密算法;(3)提出了面向隧道模式的字段灵活可选的加密机制,允许用户根据包头所含信息的重要性有选择地加密,通过减少数据的加密量获得更好的时间效率。

1　相关工作

目前学术界针对信息安全问题提出了多种安全保护机制。在安全传输架构设计方面,文献[9]提出了一种用于安全 VPLS 架构的新型快速传输机制,在保护 LAN 安全的同时减少了数据传输前的等待时间,从而大大降低了远距离客户站点间的平均数据传输延迟。文献[10]提出了一种灵活的 CP-ABE 方案,与基于短密文的 CP-ABE 方案相比,该方案可以应用于更多更一般的情况。然而该机制中的加密报文的选择基于特定的公共属性,攻击者能够根据属性特征获取加密流量的特征,用户信息存在隐私泄露的风险。文献[11]提出了一种基于正交多项式域方波混洗的选择性图像加密算法,通过配置不同参数选择不同的加密方法,充分保障了数据的机密性。然而该机制所使用的两种加密算法有着相似的计算复杂度,无法保障数据传输过程中的网络性能,不利于处理大量数据流。

在灵活加密方面,文献[12]提出了可扩展的灵活加密技术,为设备提供轻量级加密保护的同时提高加密效率。然而,该机制主要通过减少加密计算的轮次实现灵活加密,并没有改变加密复杂度,无法根本性提升加密级别。文献[13]提出了一种基于多连接传输的加密策略,利用数据本身作为加密密钥,通过多连接传输将数据封装成不同的传输路径。与传统加密策略相比,该方法减少了数据的处理时间,同时也保证了数据的安全性。文献[14]提出一种基于 P4 可编程平台的灵活可靠传输加密方案,实现了基于隐私度的数据包按需加密。然而该方案并没有利用可编程数据平面的灵活性设计多协议可用的安全传输机制,通用性较低。文献[15]提出一种动态多路径多协议加密通信机制,将高资源消耗的流量通过多条路径传输,来降低每条路径上的开销。但事实上,该机制的动态策略仅体现在对不同数据包的选择性加密,并不包含对数据包内部字段的灵活加密。

近年来,以 SDN 为代表的新型网络发展迅速,关于 P4 可编程数据平面的应用研究也层出不穷。文献[16]在数据平面中实现了加密哈希函数,以减轻针对哈希冲突的潜在攻击。文献[17]使用加扰查找表在数据层实现高级加密标准(AES)协议。文献[18]在 P4 上实现了 IEEE 802.1AE (MACsec),并引入了一种自动部署机制来为 P4 交换机之间检测到的链路提供 MACsec。在 P4-MACsec 的基础之上,文献[19]尝试在 P4 交换机中实现主机到站点的 IPsec,建立了基于 IPv4 的安全传输隧道。另一方面的研究则侧重于保护互联网用户的身份。文献[20]提出了网络元素中的监视保护(SPINE),这是一种通过隐藏 IP 地址和相关 TCP 字段(例如序列号)来实现与数据平面中的

敌对自治系统(as)进行匿名通信的系统,但是并不隐藏负载等数据信息,无法为有效数据提供安全保护。

2 设计原理与系统架构

2.1 概 述

该文基于 P4 可编程技术提出了多协议通用的安全传输机制 PIFET,应用场景和设计目标如图 1 所示。

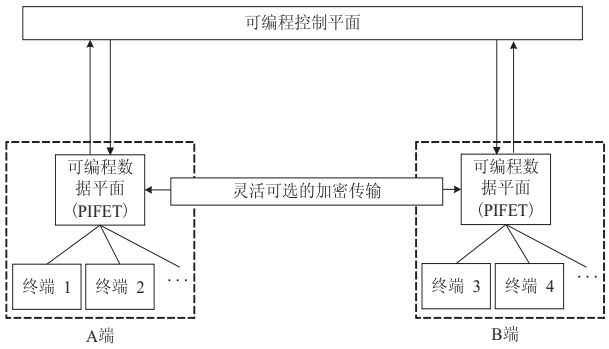

图 1 PIFET 应用场景

PIFET 运行在可编程数据平面上并通过 P4runtime 协议与可编程控制平面进行交互。部署在数据平面的 P4 交换设备允许多个终端接入,每个终端能够发送或接收多种异构协议,不同协议数据通过 P4 交机建立安全传输通道。可编程控制平面对数据平面进行统一的配置和管理,包括流表下发、密钥更新等操作。作为 PIFET 的核心原理,每一个运行 PIFET 的 P4 交换机都包含一整套完整的加解密和转发操作。该文根据异构协议的传输需要和加密需求,配置了隧道传输模式和负载加密两种安全等级不同的安全传输方法。接收设备收到数据包后按照对应的解密算法解密数据包,最后发送至接收终端。PIFET 功能的通用性极大地方便了交换设备的统一部署。

2.2 PIFET 数据平面

为了便于理解,将功能分组到功能块中,数据平面主要包含可编程解析器、安全策略功能块、加密功能块、解密功能块和转发功能块等。数据包进入转发平面后,解析器先提取报文特征然后与 MAT 中的关键字匹配,若匹配成功则执行相应动作,所匹配失败,则旁路或丢弃。图 2 描述了 PIFET 在数据平面的处理流程。

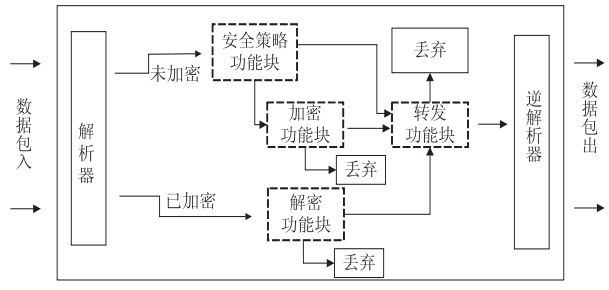

图 2 数据平面处理流程

下面对主要功能块和重要组件进行详细介绍。

(1)安全策略功能块。

安全策略功能块实现为依赖安全策略匹配表的决策模块,安全策略匹配条目由控制器下发。安全策略条目决定了属于特定数据流的数据包应该被旁路至转发功能块或是发送至加密功能块。该过程中,数据包根据源地址、目的地址以及可选的协议号、数据类型等关键信息进行匹配,并分配给两个操作之一,该模块不会丢弃数据包。

(2)加密、解密功能块。

为了扩展可编程数据平面的安全功能,利用 P4 Externs 实现了多组由特定加密算法组成的 PIFET 密码套件。加密功能块由包含密码套件的 Externs 函数和加密材料组成。每个密码套件由两个 P4 Extern 程序实现,一个用于加密,一个用于解密。隧道模式下,使用 ESP 安全协议按照控制平面下发的安全关联(Security Association,SA)信息加密、封装数据包,并通过安全参数索引(Security Parameter Index,SPI)标识与加密相关的 SA 信息。从控制器获取与原始数据包的源、目的地址所对应的隧道端点地址,并作为外部包头的一部分来封装隧道。隧道传输模式下的数据包封装方式如图 3 所示:原始数据包网络层存在多级头部,图 3 以全部加密为例展示了加密数据包的封装结构。负载加密模式则直接对所有有效载荷进行加密。

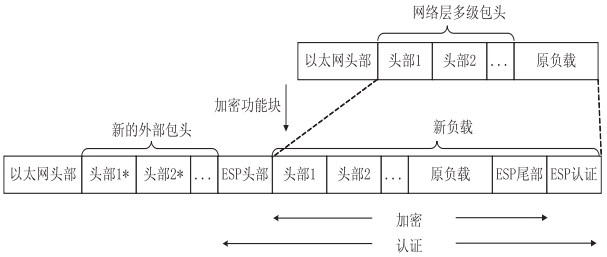

图 3 数据包加密和封装方式

(3)转发功能块。

转发功能块对接收到的数据包做转发操作。MAT 根据输入的数据类型配置不同的匹配关键字。如果有相应的匹配条目,则按照动作进行转发,否则将会被丢弃。最后数据包经转发模块发送到逆解析器,逆解析器对包头重组后从相应的出端口发出。

2.3 PIFET 控制平面

根据加密需求,该文对 SDN 控制器进行模块化设计,并通过 P4runtime 将转发和加密策略下发至数据平面,大大提升了网络的灵活性和控制能力。当数据包到达 P4 交换机时,交换机将用户的通信请求发送到控制器。根据当前网络需求,该文通过匹配控制器数据库中的映射条目,为不同类型的流量分配不同的加密策略。同时定期动态更新密钥,有效保证网络安全。

最后将所需的流表和执行动作发送到交换机。该文对SDN 控制平面进行模块化设计,包括初始化主模块、流量信息采集模块、安全策略存储模块、密钥生成和更新模块以及流规则生成和下发模块。

(1)初始化主模块。

该控制系统采用主动传递和被动传递相结合的流表下发模式。当 SDN 控制器首次连接到 P4 交换机时,控制器会针对常规的转发操作下发默认的控制流表。当控制器检测到 P4 交换机发送 Packet-in 事件时,将解析输入数据包特征,并下发生成的特定控制器流表。

(2)流量信息采集模块。

流量信息采集模块的主要功能是收集 P4 交换机上传的流量信息。当控制器检测到来自 P4 交换机的 Packet-in 事件时,将从包入中提取入站包字段。然后,控制器从入站数据包字段中提取有效信息,例如协议号、包类型、目的地址等。最后对流量信息进行处理和分析,得到用于匹配加密策略的流量分类。

(3)安全策略存储模块。

安全策略存储模块用于存储不同流量的安全需求,实现加密方法可定义。该文创建了安全策略映射数据库和地址映射数据库。安全策略映射数据库以键值对的形式存储安全策略。其中,匹配键是不同的流量类型,值是相应的加密策略。控制器判断接收流量的类型后,在本地加密策略映射数据库中查询每个流量对应的安全策略。然后,将相应的安全策略以流表的形式发送到 P4 交换机。数据平面以数据包为粒度来拆分流量,首次进入交换机且有安全需求的数据包将发往控制器获取安全方法。PIFET 通过加密来保证运行不同网络协议用户的通信安全。考虑到不同协议在安全需求,包结构以及设计初衷之间的差异,PIFET 目前设计了两种可供选择的加密方法:加密级别较高的隧道传输和安全级别较低的负载加密,同时支持用户自定义加密算法。地址映射数据库用于存储隧道传输模式下原始数据包的源、目的地址和加密后隧道端点地址的映射关系。

(4)密钥生成和更新模块。

该模块针对不同的加密方式,设计相应的密钥生成和更新方法。隧道模式下,加密密钥和 SPI 等信息均包含在 SA 中,为了配置和动态更新 SA,该文在控制器中引入安全隧道文件。安全隧道文件是在 P4 可编程交换机之间建立安全连接的基础,它包含加密材料、SA 生存期和交换机身份标识等基本信息。控制器根据安全隧道文件生成数据平面的 SA 配置数据,并通过 P4runtime 下发至交换设备。同时使用寄存器中的计数器为 SA 设置有限的生命周期以定期更新密钥。负载加密模式下,以控制器下发加密密钥,代替密钥协商过程,同时允许用户根据应用需求设置密钥更新周期。

(5)流规则生成和下发模块。

控制系统根据安全需求,构建了完整的、定义良好的流规则对象。该对象是控制器对 P4 交换机中流条目的抽象。流规则对象构造完成后,将传递给控制器中的流规则服务系统。设备驱动程序调用流表翻译子系统,并使用该系统将流规则转换成相应的表条目。然后,表条目将在 SDN 控制器的 P4Runtime 客户机中被转换成 P4Runtime 通信消息。最后,通信消息被下发到数据平面。P4 交换机根据收到的消息,向其对应的流表中添加对应的流条目。

2.4　面向异构协议的灵活加密机制

PIFET 为用户提供一种灵活,可定义的加密机制,包括加密数据类型的可定义和加密方式的可定义。由于不同类型的数据包往往有着不同的加密要求,该文设计了一种面向异构协议的灵活加密机制,允许用户在控制器的安全策略存储模块中自定义加密数据类型及其安全策略,并下发到数据平面。数据平面和控制器的主要交互流程如图 4 所示。

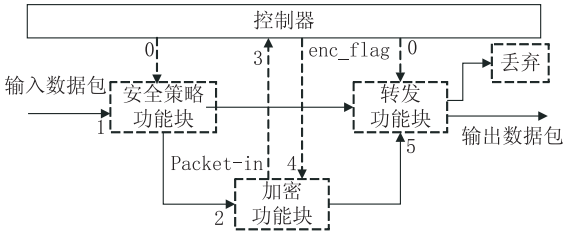

图 4　数据平面和控制器交互流程

控制器与交换机首次连接后,将默认流表下发至安全策略功能块和转发功能块。数据包进入数据平面,先匹配安全策略功能块。安全策略匹配表用于筛选符合加密条件的数据类型。如果匹配成功,则被送入加密功能块;否则送入转发模块,并与转发匹配表进行匹配。对于首次输入的数据包,加密功能块中并不包含对应的加密表项,随即触发 Packet-in 事件,将数据包发往控制器。该文引入了 8 位用户自定义元数据 enc_flag 用于标记加密策略。控制器收到数据包后,解析并查找对应的加密策略,并根据加密方式对 enc_flag 进行赋值,然后与流表一起下发并存储在加密功能块。PIFET 通过为 enc_flag 赋值区分不同的加密方法。加密功能块通过匹配头部关键字段和 enc_flag 值选择对应的加密方法。加密后的数据送入转发功能块,匹配转发或丢弃操作。用户可以修改安全策略匹配表来重新筛选有加密需求的数据包。同时,控制器根据网络流量,为流表项设置合适的空闲超时以定期更新加密策略。

2.5 包头灵活选择性加密机制

隧道传输模式下数据包的加密和封装方式如图 3 所示。该模式使用 ESP 协议对首个网络层包头(头部 1)至数据包末尾的全部字段进行加密,充分保障了负载和路由信息的机密性。然而,部分异构协议数据包含有多级网络层包头,但往往只有少数包头携带关键信息。考虑到加密过程的复杂性和系统的处理效率,若加密全部包头,必然会影响整体网络性能。为此,在隧道传输模式基础上,该文提出了包头可选择的灵活加密机制。

该机制允许用户根据网络层不同级包头所含信息的重要性,灵活选择加密起始位置。该文利用控制器对用户输入的包头名称(参数)进行散列计算并以下发流表的方式确定加密起始位置,图 5 展示了该处理流程。首先用户筛选出携带关键信息的网络层头部,然后将头部名称输入控制器。控制器收到参数后,计算散列值,并将散列值作为参数发送至数据平面的加密功能块。同时,控制器将包头名称和对应的散列值以键值对的方式保存为本地映射表。控制器每接收一个参数,都与本地映射表进行匹配,如果没有匹配条目,则计算相应的散列值并添加到匹配表中。为了能够接收控制平面下发的散列值并确定起始位置,该文在数据平面引入了包名称、散列值和偏移量相对应的三要素匹配表。解析器每解析一级包头,都按照与控制平面相同的算法计算其散列值,然后利用指针保存该包头的相对偏移量。相对偏移量记录了每一级网络层包头相对首个网络层包头的地址偏移。三元素匹配表按照索引号记录了包名称、散列值和指针值三者的映射关系。数据平面收到散列值后,先与本地三要素匹配表进行匹配,查找对应的偏移量,最后作为参数传入 Extern 中的加密函数。整个过程采用线性探测的方法避免散列冲突。

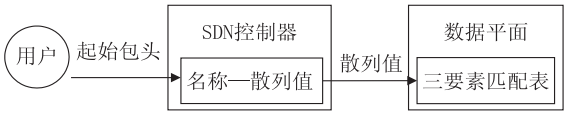

图 5 散列值的下发与匹配流程

3 实验结果与讨论

3.1 实验环境

本节对所提方案进行了实验仿真和性能分析,搭建了如图 6 所示的实验拓扑。用户通过发送设备 h1 发送异构协议数据包,接入交换机和网络交换机均为 P4 可编程交换设备。PIFET 部署在基于 PPK 的网络交换机上,PPK 是可编程数据平面的编译环境,可配置数据平面的操作,并与 P4 语言兼容。P4 交换机配备了 Ubuntu 20.04 操作系统、英特尔 i7-6700 处理器和 64 GB 内存,为异构协议数据提供安全保护。实验测试的网络性能评估基于网络性能测试工具 Spirent 和数据包分析软件 Wireshark。

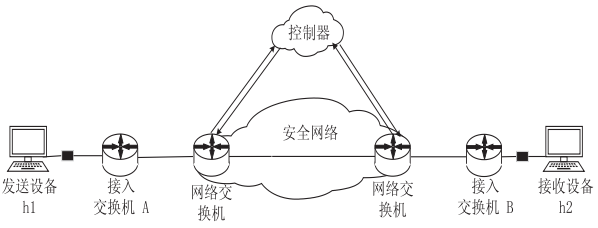

图 6 实验拓扑

3.2 网络性能测试

为了验证 PIFET 的可靠性和有效性,该文设计了两组对照实验,并设置了从 74 字节到 1 400 字节的帧长度来验证网络性能。第一组实验,选择不同的异构数据,对比了隧道模式、负载加密模式和不加密的明文传输模式三者的性能差异。隧道模式使用 SM4 国密算法加密数据包、SM3 哈希算法进行完整性校验,负载加密模式采用 DES 加密算法。为避免结果的随机性,通过计算 10 个实验结果的平均值来减小误差。经过大量的实验,从样本中随机选取 300 组数据进行网络性能分析。网络吞吐量情况如图 7 所示,网络延迟如图 8 所示。

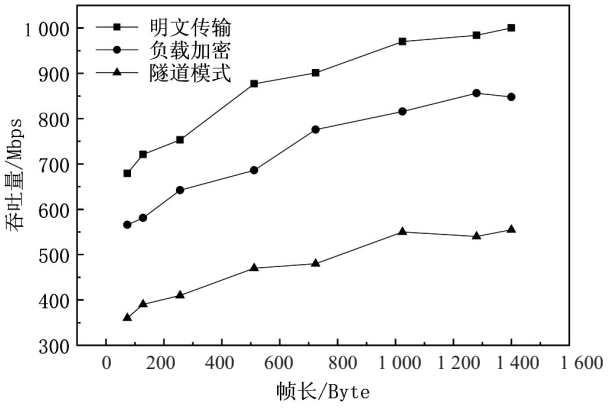

图 7 三种模式下的吞吐量对比测试

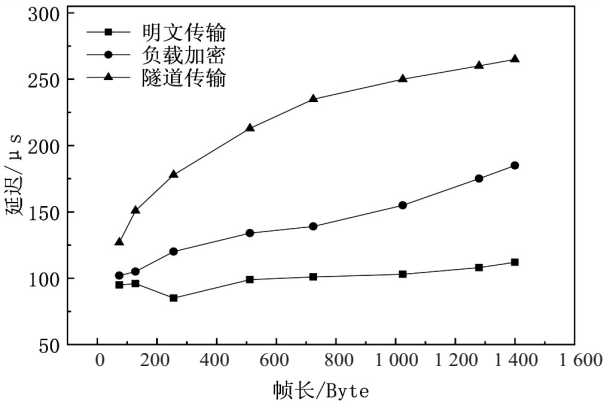

图 8 三种模式下的延迟对比测试

从图 8 中可以看出,数据包在传输过程中吞吐量随着帧长的增加而增加。不加密时的平均吞吐量是

890.48 Mbps,在三种情况中最高。基于 DES 加密算法的负载模式平均吞吐量是 723.52 Mbps,相比之下吞吐量下降了 18.7%,这是由于加密操作增加了数据在交换机中处理的时间,进一步影响接口速率,最终导致吞吐量的下降;隧道传输模式下的平均吞吐量是 472.36 Mbps,相比不加密时吞吐量降幅为 46.9%,显然隧道传输模式对吞吐量的影响高于负载加密模式。一方面是因为隧道的建立引入了额外的时延,另一方面,SM3 加密算法相比 DES 加密更加复杂,加密时间的延长进一步提高了丢包率。

从图 8 可以看出,网络延迟随着数据帧长度的增加而增加,其中隧道模式的延迟最高,这与该模式下数据包的封装操作和加密算法高复杂度密不可分。不加密时,网络延迟随着帧长的增加变化不大。负载加密情况下,延迟随着帧长的增长显著上升,这是由于长包导致负载加密的时间随之增加。隧道传输模式下,短数据包的延迟增长率大于长包。这是由于在建立隧道的过程中封装了新的外部包头,从而导致包长度变长,而这对短数据包的影响更为明显。从上述实验结果可以看出,在通信过程中,较高安全级别的加密算法所使用的密钥更为复杂,在为数据包带来高机密性的同时,牺牲了部分通信性能。

第二组实验选择网络层有多级头部的异构协议测试基于包头灵活选择性加密机制的隧道模式(简称自定义隧道模式)所带来的网络性能增益。以 MF 和 Geo 协议为例,MF 数据包在网络层分为包含 MF_type 字段的一级包头和包含源、目的信息的二级包头。MF_type 标明该 MF 的报文类型,不同类型的数据包功能不同。由于一级包头不含关键路由信息,所以该实验选择从二级包头开始加密;Geo 报文在网络层含有三级包结构,分别是基本包头,公共包头和可选包头。所有 Geo 数据包的基本包头和公共包头都有着相同的结构,主要用来标识版本号和可选包头的类型。而可选包头存储了详细的源、目的位置信息,并决定着 Geo 报文的功能,所以在该实验选择从三级包头开始加密。同时,在相同环境下添加了加密所有网络层包头的传统隧道模式作为对照实验。该实验对 MF 协议和 Geo 协议分别进行了 5 组测试,最后计算 10 组实验结果的平均值。网络吞吐量和延迟对比如图 9 和图 10 所示。

从图 9 可以看出,自定义隧道模式的吞吐量优于传统隧道模式。当帧长较短时,相比传统隧道模式,由于自定义隧道模式中省去的加密字段占整个报文的比例更大,时间效率的提升比例更高,所以吞吐量的改善效果较为明显。然而,当帧长增加时,未加密字段所占比例降低,吞吐量优化效果下降。从图 10 来看,自定义隧道模式在延迟方面相对传统隧道模式有所改善,平均降低 14 微秒,单包延迟改善效果并不明显。然而,当 PIFET 处理连续、高速率的数据流时,得益于多个数据包延迟优化的累积效应,相比传统隧道模式,使用自定义隧道模式加密的数据流的延迟得到明显改善。

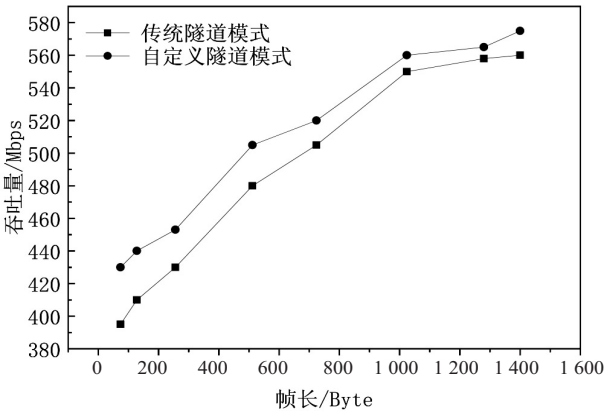

图 9　两种模式下的吞吐量对比测试

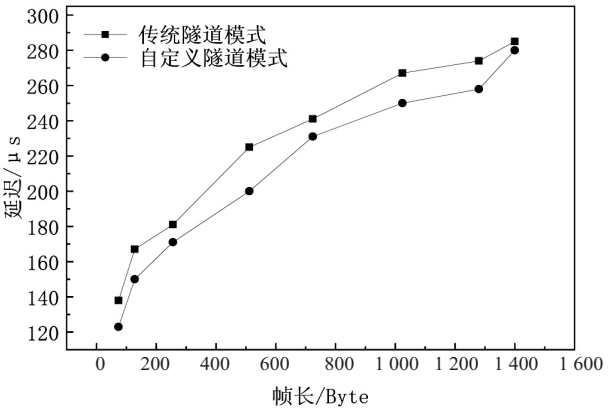

图 10　两种模式下的延迟对比测试

4　结束语

针对异构协议安全机制不完善,数据传输过程中面临隐私泄露等安全问题,该文提出了协议无关的灵活加密传输机制(PIFET)。PIFET 基于可编程数据平面,实现了多协议解析,为异构数据提供通用的安全传输机制。同时,提出了灵活的加密机制,允许用户根据安全需求自定义加密方法和加密算法,提高了加密的灵活性。另外,PIFET 实现了字段灵活可选的加密机制,通过筛选并加密含有关键信息的包头字段,提高系统的时间效率。最后,在可编程数据设备上验证了 PIFET 的有效性和系统性能。后续工作将设计链路状态的检测算法,并依据链路状态适时调整加密策略。

参考文献:

[1] 胡宇翔,伊　鹏,孙鹏浩,等. 全维可定义的多模态智慧网络体系研究[J]. 通信学报,2019,40(8):1-12.

[2] JACOBSON V,SMETTERS D K,THORNTON J D,et al. Networking named content[C]//Proceedings of the 5th in-

ternational conference on emerging networking experiments and technologies. Rome: ACM, 2009: 1-12.

[3] RAYCHAUDHURI D, NAGARAJA K, VENKATARAMANI A. Mobilityfirst: a robust and trustworthy mobility-centric architecture for the future internet[J]. ACM SIGMOBILE Mobile Computing and Communications Review, 2012, 16(3): 2-13.

[4] PERRIG A, SZALACHOWSKI P, REISCHUK R M, et al. SCION: a secure Internet architecture[M]. Berlin: Springer, 2017.

[5] SANDONIS V, SOTO I, CALDERON M, et al. Vehicle to Internet communications using the ETSI ITS GeoNetworking protocol[J]. Transactions on Emerging Telecommunications Technologies, 2016, 27(3): 373-391.

[6] JIANG Baohua, SUN Hui. Analysis of computer network security and countermeasures based on big data[C]//2019 international conference on information science, medical and health informatics (ISMHI 2019). Paris: Information Technology, 2019: 203-206.

[7] SU K, BRONZINO F, RAMAKRISHNAN K K, et al. MFTP: a clean-slate transport protocol for the information centric mobility first network[C]//Proceedings of the 2nd ACM conference on information-centric networking. New York: ACM, 2015: 127-136.

[8] LEE J H, ERNST T. Overhead analysis on secure beaconing for GeoNetworking[C]//2011 IEEE intelligent vehicles symposium (IV). Baden: IEEE, 2011: 1049-1053.

[9] LIYANAGE M, YLIANTTILA M, GURTOV A. Fast transmission mechanism for secure VPLS architectures[C]//2017 IEEE international conference on computer and information technology (CIT). Helsinki: IEEE, 2017: 192-196.

[10] JIANG Y, SUSILO W, MU Y, et al. Flexible ciphertext-policy attribute-based encryption supporting AND-gate and threshold with short ciphertexts[J]. International Journal of Information Security, 2018, 17(4): 463-475.

[11] KRISHNAMOORTHI R, MURALI P. A selective image encryption based on square-wave shuffling with orthogonal polynomials transformation suitable for mobile devices[J]. Multimedia Tools and Applications, 2017, 76(1): 1217-1246.

[12] MEDILEH S, LAOUID A, EULER R, et al. A flexible encryption technique for the internet of things environment[J]. Ad Hoc Networks, 2020, 106: 102240.

[13] ZHOU S, LIN R, ZOU H, et al. An encryption algorithm based on multi-connection transmission[C]//2017 IEEE 17th international conference on communication technology (ICCT). Chengdu: IEEE, 2017: 1296-1300.

[14] QIN Y, QUAN W, SONG F, et al. Flexible encryption for reliable transmission based on the P4 programmable platform[C]//2020 information communication technologies conference (ICTC). Nanjing: IEEE, 2020: 147-152.

[15] ZHU S, CHEN D, YANG M, et al. Dynamic multi-path and multi-protocol encrypted communication mechanism[C]//2021 IEEE 13th international conference on computer research and development (ICCRD). Beijing: IEEE, 2021: 58-62.

[16] SCHOLZ D, OELDEMANN A, GEYER F, et al. Cryptographic hashing in P4 data planes[C]//2019 ACM/IEEE symposium on architectures for networking and communications systems (ANCS). [s. l.]: IEEE, 2019: 1-6.

[17] CHEN X. Implementing AES encryption on programmable switches via scrambled lookup tables[C]//Proceedings of the workshop on secure programmable network infrastructure. New York: SPIN, 2020: 8-14.

[18] HAUSER F, SCHMIDT M, HÅBERLE M, et al. P4-MACsec: dynamic topology monitoring and data layer protection with MACsec in P4-based SDN[J]. IEEE Access, 2020, 8: 58845-58858.

[19] HAUSER F, HÅBERLE M, SCHMIDT M, et al. P4-IPsec: implementation of IPsec gateways in P4 with SDN control for host-to-site scenarios[J]. arXiv: 1907.03593, 2019.

[20] DATTA T, FEAMSTER N, REXFORD J, et al. {spine}: surveillance protection in the network elements[C]//Proceedings of the 9th USENIX workshop on free and open communications on the internet (FOCI 19). Austin: WikiCFP, 2019: 1564-1571.