

一种基于零信任的SDN访问控制架构

梁 赓, 韩培胜, 赵世海

(信息工程大学, 河南 郑州 450001)

摘 要: 针对软件定义网络(SDN)缺乏有效认证方式与持续访问控制能力的问题, 提出一种基于零信任的SDN访问控制架构。首先, 该架构以零信任思想为指导, 采用验证属性签名的方式进行身份认证, 并在用户访问过程中持续认证; 其次, 通过信任度量的方式控制用户访问, 建立用户历史行为模型检测内部威胁, 根据用户信任等级授予相应访问权限。实践表明, 该架构为SDN提供了先认证、后连接的安全接入模式, 增强了SDN检测内部威胁的能力。

关键词: 软件定义网络; 零信任; 属性签名; 信任度量; 访问控制

DOI: 10.11907/rjdk.241977

中图分类号: TP393

文献标识码: A

文章编号: 1672-7800(2026)003-0157-08

扫描二维码阅读全文:



A Zero Trust Based SDN Access Control Architecture

LIANG Geng, HAN Peisheng, ZHAO Shihai

(Information Engineering University, Zhengzhou 450001, China)

Abstract: A zero trust based SDN access control architecture is proposed to address the lack of effective authentication methods and persistent access control capabilities in Software Defined Networking (SDN). Firstly, the architecture is guided by the zero trust concept and uses authentication attribute signatures for identity authentication, with continuous authentication during user access. Secondly, user access is controlled through trust metrics, and a user history behavior model is established to detect internal threats, granting corresponding access permissions based on user trust levels. Practice has shown that this architecture provides SDN with a secure access mode of authentication before connection, enhancing SDN's ability to detect internal threats.

Key Words: software defined networking; zero trust; attribute signature; trust measurement; access control

0 引言

软件定义网络(Software Defined Networking, SDN)通过集中式的控制器管理网络, 改变了传统网络通过路由器和交换机控制网络的模式, 因此应用前景广阔^[1]。然而, SDN存在一系列安全问题: ①缺乏有效的身份认证方式, SDN交换机转发过程容易受非法数据包干扰, 面临外部安全威胁^[2]; ②缺乏有效的访问控制机制, 难以及时发现并响应内部安全威胁。

为此, 文献[3-4]介绍了SDN环境下DoS、DDoS攻击的背景, 系统回顾、整理了解决DoS、DDoS攻击的现有方案, 并将其分为: 基于统计的技术、基于机器学习的技术、神经网络和其他检测方法, 还指出了各种检测技术的优缺点, 提出了解决DoS/DDoS攻击的未来研究方向。

文献[5]提出在SDN上部署虚拟节点, 利用随机森林和自编码器检测僵尸网络, 虽能有效检测防御Crossfire攻击, 但仅针对单类网络攻击, 扩展能力不足。文献[6]根据流表特征来检测DDoS攻击, 先使用主成分分析优化流表特征, 再使用支持向量机来检测攻击。文献[7]提出在SDN环境中使用属性签名验证的方法处理用户数据包, 采用多个控制器管理、控制网络, 以处理异常数据包, 解决了单一控制器失效的问题, 但未研究可能出现的内部安全威胁。文献[8]提出了SDN环境下的轻量级数据包转发验证方案, 采用Packet-in消息机制便于控制器监控、处理数据包, 实现了灵活和智能的网络管理, 但该方案仅验证用户数据包, 未控制用户访问过程, 因此无法防御内部攻击。文献[9]提出一个基于区块链的安全框架, 集成了定制的算法认证、加密和访问控制, 降低了SDN的虚假数据注入、中间人攻击和未授权访问等风险。

收稿日期: 2025-02-19

作者简介: 梁赓(2001-), 男, 信息工程大学硕士研究生, 研究方向为网络安全; 韩培胜(1978-), 男, 博士, 信息工程大学副教授、硕士生导师, 研究方向为网络安全和人工智能; 赵世海(1990-), 男, 信息工程大学硕士研究生, 研究方向为软件测试、人工智能。

零信任要求“从不信任,始终验证”,主张网络中的每个连接都经过身份验证,强调对每个访问行为进行持续验证和动态授权,用户基于最小权限原则访问资源^[10]。目前,许多学者在SDN中引入零信任模型,以提升SDN应对网络安全威胁的能力^[11]。其中,SDN将控制面和数据面分开,提供了网络流量的动态控制能力,与零信任的动态认证和细粒度访问控制相符^[12];SDN的集中式控制和可编程性使网络能根据业务需求快速调整,从而实现动态的零信任策略部署。

文献[13]提出SDP-SDN一体化架构,可抵御多种网络攻击,主要利用现有软件定义边界(Software Defined Perimeter, SDP)组件(SDP客户端、网关等)来实现零信任认证。然而,该架构的网络组件较多,需要协调各组件以保证其兼容性。文献[14]提出一个基于智能零信任的SDN安全框架,利用深度学习算法检测流量异常,通过零信任架构保护网络资源以提升SDN的安全性。文献[15]基于零信任模型,利用SDP框架为电力物联网提供一种敏感数据访问控制方法。文献[16]基于SDP架构提出一种零信任访问方案,通过密钥协商分发密钥、指定验证者签名来验证访问者身份,可有效防御敲门攻击、身份盗用等网络攻击。文献[17]提出一种基于SDN的物联网边缘节点之间数据流零信任的管理方法,可转发验证轻量级的数据包,以检测数据流传输链路中的恶意交换节点。

当前,SDN的零信任访问控制,有的通过添加外部认证服务器进行身份认证;有的依靠SDP技术实现访问控制,即将SDP组件添加到SDN,协调各组件实现零信任访问控制^[18]。然而,这些方式在实施过程中涉及的组件多、管理难度大,SDN控制器的负载和兼容性难以得到保障。

控制器是整个SDN网络的核心,负载的增加可能会导致控制器处理速度下降甚至失效^[19]。本文为了解决该问题,提出基于零信任的SDN访问控制架构,不使用SDP等其他外部组件,基于SDN现有结构,以零信任思想为核心,通过定义和编排安全策略来实现零信任访问控制。首先,利用P4交换机实现基于属性签名的身份认证;其次,开发SDN应用对用户进行信任度量,从而实现动态访问控制。

1 系统架构

1.1 架构组成

传统的SDN包括应用层、控制层和数据层。应用层为用户提供高级的网络服务和应用,允许自定义网络策略;控制层负责网络的逻辑控制、状态监控等;数据层负责转发数据包、提供标准设备接口等。控制层可通过数据层提供的接口与各种网络设备通信,进行设备协调和扩展。通过SDN控制器可快速部署和修改安全策略,精确控制网络流量,实现零信任的访问控制。SDN应用层支持用户自主开发,结合人工智能技术能更有效地检测、响应潜在的安

全威胁。

本文架构如图1所示,主要模块包括:①SDN控制器:负责管理和控制网络,依据用户信任值授予用户相应访问权限,通过下发流表规则控制交换机的转发行为;②SDN交换机:根据流规则处理数据包(转发、丢弃等);③属性签名认证组件:包括属性签名添加模块、属性签名认证中心和P4交换机,P4交换机为SDN提供了数据转发平面的可编程性,能实现属性签名认证功能;④信任度量系统:包括日志数据库、用户行为分析应用、信任管理应用等,负责根据用户属性信息和历史信任值计算初始信任值;⑤根据用户历史行为基线模型:检测访问过程中的异常行为,并计算异常分数,持续更新用户信任值。

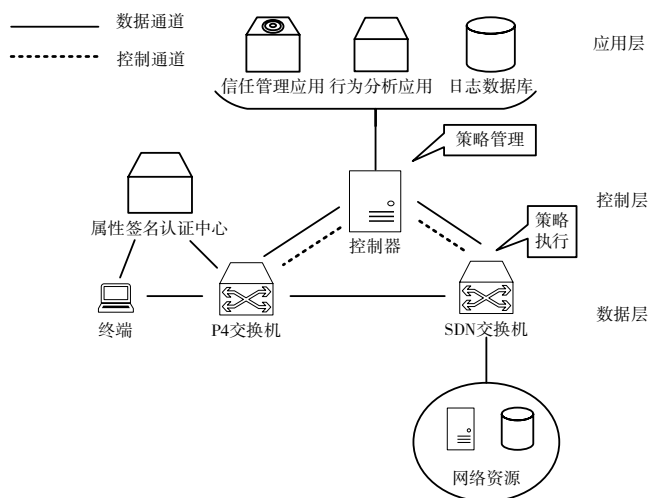


Fig. 1 SDN access control architecture based on zero trust

图1 基于零信任的SDN访问控制架构

1.2 用户访问过程

以用户访问服务器资源为例,访问过程如图2所示。在接入网络前,首先用户登录终端应用生成认证数据包,并将数据包发送至P4交换机进行身份认证;其次,将认证结果上报SDN控制器;最后,认证通过后被允许接入网络,并在数据报文中加入属性签名标识,信任度量系统通过计算信任等级,授予相应的访问权限。

在用户访问过程中,检测到异常行为后,系统开始计算异常分数,更新信任等级。当信任等级降低后,控制器立即调整用户的访问权限。整个访问过程中,P4交换机会随机抽取用户数据包进行身份校验,以验证数据的流合法性,如果出现非法数据包,将立即停止用户访问网络。

2 基于属性签名的身份认证

属性签名除了能表征用户身份之外,还具备密码加密和标签标识的属性。将用户身份、设备等属性生成的属性签名标识添加到用户数据包中,可提供安全的数据流转发和细粒度控制。本文方案的安全性基于以下假设:①SDN交换机和控制器安全;②数据平面与控制平面间的通信安

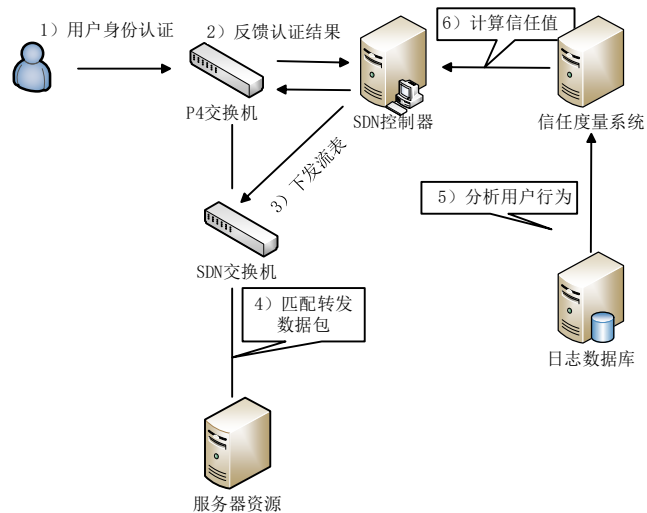


Fig. 2 User access process
图 2 用户访问过程

全;③属性签名认证中心与属性签名添加应用、P4交换机之间通过TLS信道进行通信。

2.1 属性签名方案描述

在传统密码身份认证中,通信方式是一对一。如果将其用到SDN中,因SDN的分布式特性,容易出现密钥管理和分配复杂等问题,通信开销较大。目前,数字签名在操作系统等应用场景的数据合法性校验中被广泛使用。

属性签名与数字签名类似,通信方式为一对多,发起者用自己的属性集进行签名操作,并将属性集发给验证者。验证者通过比较接收的属性集与发起者声称的属性集,如果一致则认可该签名结果,否则认为该签名为假。由于属性签名将用户身份特征进行了细致划分,因此灵活性较高,当系统要更新访问控制结构时,仅需改变访问属性即可实现^[20]。此外,属性签名方案可提供多元化的安全方式,符合SDN网络身份认证和报文安全转发的需求。

为此,本文所提方案由验证模块生成访问控制树,发起者使用自身属性对数据包签名并加密传输,验证模块负责解密签名信息,然后比较发起者的属性是否满足访问控制树。其中,访问树结构T中存储着用户属性集合,当用户满足其属性条件时,表示该用户的属性签名合法,否则验证失败。在图3所示的访问树中,管理员申请进行写操作,由于达到了访问树的条件则验证成功;一旦不满足条件,则签名验证失败。

如图4所示,本文将属性签名标识放置在IP首部的Option字段中。其中,Option字段由属性集合与属性签名两部分组成,分别占据8字节和32字节;属性集合字段中存储了用户执行属性签名所需的全部属性信息,P4交换机数据包解析模块将该字段解析后进行签名验证;属性签名字段则保存用于属性签名的具体签名数据。

2.2 属性签名认证流程

如图5所示,属性签名认证结构主要包括:

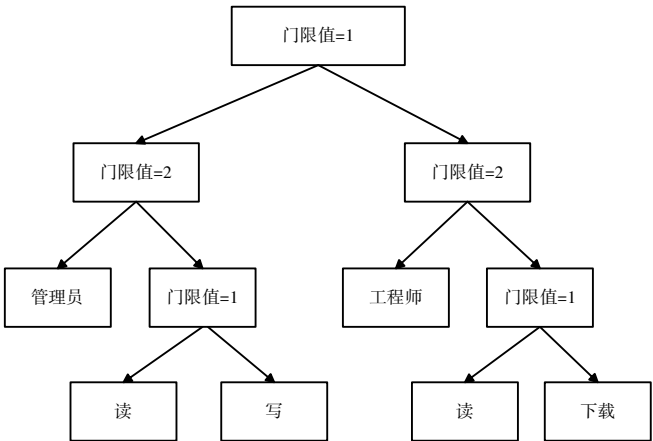


Fig. 3 Access tree example
图 3 访问树示例



Fig. 4 IP header with attribute signature identification
图 4 带属性签名标识的IP首部

(1)访问用户。每个用户拥有唯一的属性信息(包括用户名、设备等)标识。

(2)属性签名认证中心(简称为认证中心)。整个系统所需的参数和签名过程中使用的密钥都由认证中心所产生。

(3)属性签名添加应用。以应用方式安装在用户终端,根据用户信息集成属性集,并使用属性私钥为数据包添加签名标识。

(4)P4交换机。通过P4语言进行编程,可自定义配置和编程,在P4交换机中的开发模块用于属性签名认证。其中,属性标识解析模块负责解析用户认证数据包;身份认证模块负责对用户属性信息进行认证,并将认证结果上报控制器;采样控制模块负责根据控制器命令,对用户数据包进行随机采样、身份校验,该过程不会影响数据流的正常转发;数据转发模块可根据属性标识转发IP分组。

(5)SDN控制器。接收P4交换机的认证结果,并将流表下发至P4交换机和转发交换机。

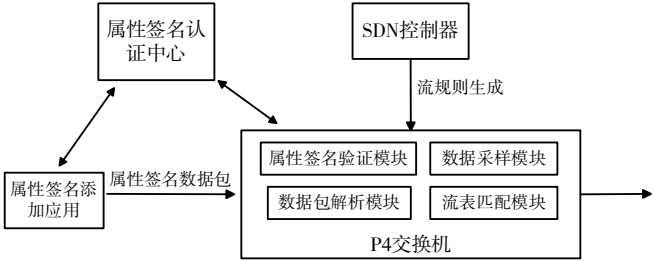


Fig. 5 Attribute signature authentication structure
图 5 属性签名认证结构

图6为属性签名认证实现过程。首先,认证中心产生签名所需的公开参数;其次,P4交换机中的属性签名认证模块检查签名认证条件;再次,生成访问树T后传送到认证中心;最后,认证中心接收后,为T生成所需参数,并发

送回签名认证模块。在此过程中,属性标识添加应用为用户数据报文添加属性标识;P4交换机和转发交换机按属性标识转发用户数据包;控制器随机要求P4交换机抽样用户数据包进行解析,以验证数据包合法性。

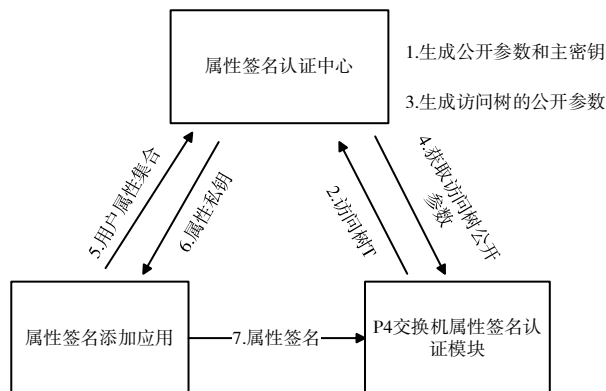


Fig. 6 Attribute signature process

图6 属性签名过程

由图6所示,用户打开属性标识添加应用以收集用户属性信息,并将信息形成属性集,发送给认证中心服务器。中心根据信息生成对应的属性私钥,传输至标识添加应用。应用接收到私钥后生成签名标识,将其嵌入至IP数据包首部,并重新封装数据包,发送给P4交换机。P4交换机收到认证数据包后,通过偏移字段解析分组头,将分组解析成IP地址、IP协议类型等交换机可识别的字段。解析完成后,验证属性签名字段,并将认证结果上报SDN控制器。认证通过后,允许用户接入网络,否则默认丢弃该用户数据包,拒绝其接入网络。

3 动态访问控制

本文采用基于信任的动态访问控制模型对用户进行动态管理,将定量的行为数据映射为信任等级^[21]。首先,建立用户历史行为基线模型;其次,依据用户当前属性计算初始信任值;最后,根据用户行为偏离基线模型的偏离度计算异常得分,以动态更新信任值。其中,用户信任值大小为 $[0, 100]$,信任值对应信任等级,依据信任等级授予访问权限。本文将网络资源分为普通资源和重要资源,依据信任值可将信任等级划分为3类: $[0, 70)$ 为完全不信任,无法访问网络资源; $[70, 90)$ 为普通信任,可访问普通资源; $[90, 100]$ 为高度信任,可访问重要资源。

3.1 访问控制模型的建立

访问控制模型通过用户行为建模识别异常行为,从而实现访问控制。由于建模用户行为的前提是分析用户行为,但用户在网络中的行为复杂多样,包括资源访问行为、设备使用行为、通信行为、登录行为等^[22]。行为分析的关键是采集数据和提取特征,对于单个用户而言,从用户节点到相应终端节点路径上产生的一系列数据,即其历史行为数据。

本文收集用户在访问过程中的日志数据,以用户身份作为划分依据,为每个用户建立专属的文件,按时间顺序存放其所有日志数据。数据获取之后,先将数据进行预处理,再提取特征建立行为模型。由于识别异常行为的准确性,与行为模型是否有效密切相关,目前行为建模的方法包括:基于规则的方法、基于统计的方法、机器学习方法等^[23]。其中,机器学习方法包括单类支持向量机、自编码器等传统方法,也包括深度学习方法。针对异常行为识别中,传统机器学习方法部署和调试较为简单、分类效果好,但用户在网络中的访问流量和日志信息数据庞大且复杂,在特征提取和数据分析方面的难度高、耗费资源多。为此,本文尝试利用神经网络降低数据维度、自动提取特征,将深度学习与传统机器学习结合起来识别异常行为。

支持向量数据描述(Support Vector Data Description, SVDD)是单类支持向量机的一种实现方式,建模简单、可解释性强、泛化能力强,建立的基线模型表达能力、识别效果较好。然而,该方法需要大量的存储内存,难以处理复杂高维数据。为此,本文采用一种深度支持向量数据描述(Deep SVDD)的方法来建立基线模型,如图7所示^[24]。该方法使用深度神经网络学习数据的非线性映射,因此数据表示能力更强,既减少了运算所需的存储内存,还能提升模型运算效率。

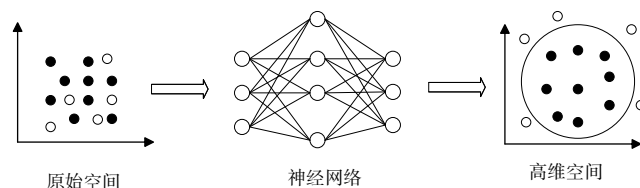


Fig. 7 Deep SVDD model

图7 Deep-SVDD模型

3.2 信任值度量

为了全面计算用户信任值,采用综合的信任度量方法,在初始信任值基础上,根据用户行为异常程度动态更新信任值。同时,为了解决盗用账号或恶意用户通过长期稳定的表现来提升信任值,以获取更高的访问权限对网络造成危害的问题,用户信任值由网络管理者处理,信任度量系统只根据异常行为减少信任值。

综上,本文将信任值的计算描述为初始信任值计算和动态信任值计算。

(1)初始信任值计算。用户访问网络的数据包中携带用户属性标识信息,通过属性标识可检查主体和环境的安全性,便可计算出用户的初始信任值。其中,主体是指访问的用户;环境是指用户的登陆设备和网络参数。主体信任值如式(1)所示。

$$st_i = \alpha_1 T_i + \alpha_2 (1 - p) \quad (1)$$

式中: st_i 表示主体信任值; α 表示评价参数权重, $\alpha_1 + \alpha_2 = 1$; p = 密码错误次数/允许错误总次数,表示密码输入正确率; T_i 表示用户历史信任值。

通过数据包属性标识包含的设备内存占用率和网络带宽占用率,即可计算环境信任值,如式(2)所示。

$$et_i = \sum_{j=1}^2 \beta_j E_{ji} \quad (2)$$

式中: et_i 表示环境子属性; β_j 表示对应的权重, $\sum \beta = 1$; E_{ji} 为网络带宽占用率; E_{2i} 为内存占用率。

用户的初始信任值等于两项信任值的加权求和,如式(3)所示。通过多次实验可评估出最合适的优化权重参数,再将 t 转化为整数,即可表示用户的基础信任度。

$$t = \mu_1 st_i + \mu_2 et_i \quad (3)$$

式中: μ_1, μ_2 为权重参数,由网络管理员设定,且 $\mu_1 + \mu_2 = 1$ 。

(2) 动态信任值计算。本文采用深度支持向量机(Deep SVDD),联合训练神经网络与支持向量机(SVDD),在特征空间中找到一个包含全部或大部分数据的最小超球,并通过神经网络将数据映射到超球内。该方法结合了深度学习与支持向量数据描述算法,将数据特征与单类分类目标放在同一个损失函数中进行学习,以最大程度学习有用的数据特征,以达到分类目标。

本文定义 α 表示超球体球心,代表正常行为数据的中心; R 表示超球体半径,代表正常行为数据分布的范围;样本到球心的距离表示每个训练样本与超球体球心的距离。通过 Deep SVDD 建立的历史行为基线模型识别异常点,并计算异常分数,将异常分数转化为相应整数,即可在初始信任值基础上更新信任值。

首先,定义一个特征映射函数 $\phi(x, W)$,使用神经网络参数 W 将输入数据 $x \in X$ 映射到特征空间 F 。

$$\phi(x; W): X \rightarrow F \quad (4)$$

其次,依靠球心 α 、半径 R 定义超球体大小优化函数,以训练 SVDD 模型,包括最小化超球体的体积和惩罚位于超球体外的点。

$$\min_{R, W} \left\{ R^2 + \frac{1}{\nu n} \sum_{i=1}^n \max \{0, \|\phi(x_i, W) - \alpha\|^2 - R^2\} + \frac{\lambda}{2} \sum_{l=1}^L \|\phi(W^l)\|^2 \right\} \quad (5)$$

式中: R^2 为超球体的体积,要尽可能小;惩罚项 ν 为一个超参数,用于平衡边界的宽松程度,如果样本点到球心的距离大于 R ,则该点被视为异常,会受到惩罚;网络权重的正则化项 λ 为正则化系数,用于防止模型过拟合。

对于测试样本 x ,异常分数 $s(x)$ 用于评估测试样本是否可能是异常,定义为该样本在特征空间中到超球体中心 α 的距离的平方,计算公式如式(6)所示。

$$s(x) = \|\phi(x; W^*) - \alpha\|^2 \quad (6)$$

式中:如果 $s(x) \geq \alpha R^2$ 则样本 x 被认为是异常; αR^2 作为检测的阈值,参数 α 可在模型训练优化中获取,也可以由架构管理员人为设定。

超球体中心 α 在特征空间中通过计算训练样本的平均值得到,超球体半径 R 通过优化过程确定。

$$\alpha = \frac{1}{n} \sum_{i=1}^n \phi(x_i, W) \quad (7)$$

式中:当 $s(x) \geq \alpha R^2$ 时判定为异常行为; $r = s(x) - \alpha R^2$,用来衡量异常点偏离检测阈值的程度,值越大表示偏离程度越大。

综合信任值的计算公式如式(8)所示。首先,在计算出初始信任度后给出用户当前信任值;其次,根据用户在网络中的访问行为,持续检测用户异常行为,动态更新信任值;最后,将 r 值转换为整数,得到用户的信任值更新公式。

$$\text{综合信任值}(T) = \text{基础信任值}(t) - r \quad (8)$$

4 实验结果与分析

4.1 实验环境

本文使用联想 P720 工作站进行实验,配置为: Intel Xeon Silver 4210R CPU, 128 GB 内存, RTX A5000 显卡。操作系统为 Ubuntu 20.04.1, 使用 Mininet 环境进行实验模拟,其他软件和驱动程序均已安装。首先,验证属性签名认证方式的检测准确率和验证开销,以验证架构的安全接入能力;其次,验证模型检测内部威胁的效果,以体现架构应对用户内部威胁的能力。

4.2 基于属性标识的报文验证实验

图8为实验拓扑。通过 P4 编程生成 JSON 格式描述文件,再导入 BMv2 中实现所需功能。在主机 h1 上利用自定义发包脚本程序向主机 h2 发送数据流,通过抓包工具 Wireshark 在主机 h2 处抓取数据报文进行分析。以 0.5 的概率发送数据流,该数据流中属性签名字段存在伪造和篡改,共发送 100 次,每次数据流包括 1 000 个数据包,通过实验结果的平均值来计算漏报率,以表征检测准确率。

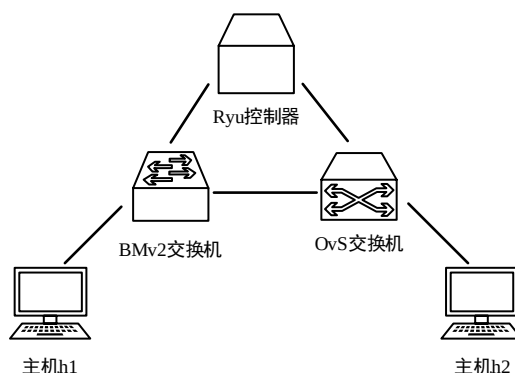


Fig. 8 Experimental topology

图8 实验拓扑

本文将检测因子 θ 分别取 10、7、3 进行实验。当系统检测发现非法数据包并上报控制器,则认为检测成功,否则视为漏报,如图9所示。由此可见,随着检测因子减小漏报率降低,表明更高的采样比能带来更高的安全性,但采样比越高 CPU 的使用率也越高,系统开销随之增大。检测因子为 10、7、3 时, CPU 的平均使用率分别为 8.2%、

9.7%、15.3%。为了在保证检测准确率较高的同时,不增加过多的系统开销,将检测因子设置为7。

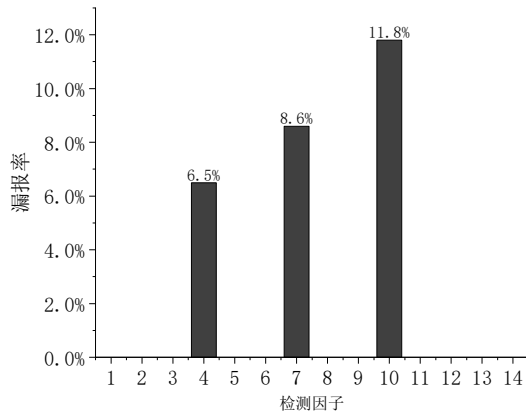


Fig. 9 Missing reporting rate

图9 漏报率

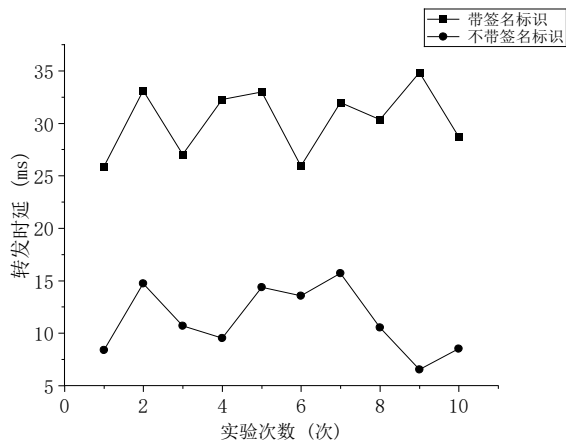


Fig. 10 Forwarding delay

图10 转发时延

接下来,将检测因子数值设为7,发送1 000个数据包(一半数据包包含签名标识,另一半不含签名标识)重复实验50次,比较交换机执行两种数据包的转发时延,以检验带有签名标识的数据包转发时的额外开销。本文实验分别统计了10次转发时延,每次时延为每发送5次数据流计算得到的平均值,结果如图10所示。由此可见,没有属性签名标识的数据包的平均转发时延为11.25 ms,包含签名标识的数据包的平均转发时延为30.30 ms,原因为有部分数据包被用在P4交换机属性签名认证模块中进行身份认证,该过程涉及多个部件间的通信,导致了架构的转发时延增大。然而,实验数据表明架构转发时延虽增大了一些,但开销在主机规模中等的网络中仍然适用。

文献[9]的转发验证方案基于用户身份进行签名验证,通过控制器进行身份认证,随着用户数量增大,控制器可能会出现失效的情况。本文基于用户属性进行签名认证细化了用户属性,灵活性更强。比较实验开销发现,文献[9]转发时延数值为33.17 ms,本文方法为30.30 ms,虽然本文方法采用了随机抽样数据包进行验证,验证开销有所增加,但系统的整体转发时延增加并不明显,还低于文

献[9]的方法,证实了本文方法在进行属性签名认证时,在保证较高准确率的同时,系统开销仍在可接受范围内。

4.3 内部威胁检测实验

实验利用Deep SVDD算法训练生成用户行为基线模型,在训练时将数据集中的正常用户行为数据集集成到Ryu控制器中,使用带有异常行为的用户数据进行测试,以测试控制器检测用户异常行为的能力。

目前,训练模型使用的是CERT-IT数据集,该数据集通常被用来进行内部威胁检测实验,数据来自真实企业环境,记录了众多用户行为:用户登录时间、注销时间、使用设备、文件访问等,还包含了某些用户的异常行为。CERT-IT数据集中有用户、设备两类终端实体,分别用A、B表示;还有登陆注销、硬件连接、访问网站、收发邮件、文件使用5类行为实体,分别用C、D、E、F、G表示,标签集合为{A,B,C,D,E,F,G}。

为了符合实验需求,将数据集进行归集、筛选,依据数据集实体及其属性提取部分特征来训练用户行为模型。本文选择数据集中的恶意用户DNS1758进行实验,由数据集标签可知,该用户前90天内的数据不含恶意行为,因此将这90天的数据作为训练数据,将后410天的数据作为测试数据,模型训练完成后集成到Ryu控制器中。

在内部威胁检测实验中,通常用查准率(P)、查全率(R)、F1得分(F1)等指标来评价模型性能,如式(9)一式(11)所示。

$$P = \frac{TP}{TP + FP} \quad (9)$$

$$R = \frac{TP}{TP + FN} \quad (10)$$

$$F1 = \frac{2 \times R \times P}{P + R} \quad (11)$$

在二分类实验中,通常将样本的实际类别与模型预测的类别进行比较,包括以下4种情况:真正例(TP)、假正例(FP)、真反例(TN)和假反例(FN),这些指标分别代表各自类别的样本数量,所有样本的总和等于这4个指标的和,分类结果可通过用混淆矩阵(见表1)进行表示。

Table 1 Confusion matrix

表1 混淆矩阵

真实情况	预测结果	
	正例	反例
正例	TP(真正例)	FP(假正例)
反例	TN(真反例)	FN(假反例)

图11为用户在第210天行为检测结果。由此可见,该用户在第210天的访问行为中存在3条异常行为,涉及文件窃取和身份盗用,3条异常行为均被检出,未出现误检、漏检。

此外,将本文所提方法与文献[25]的方法在相同的数据集下进行比较,以验证所提方法的优势,实验结果如表2所示。文献[25]首先采用深度特征合成(Deep Feature Synthesis, DFS)算法提取用户行为特征;其次,使用主成分

分析(Principal Component Analysis, PCA)进行降维;最后,采用单类支持向量机或随机森林算法进行异常检测。

由表 2 可知,本文所提方法查准率、查全率、F1 得分较高,证明了该方法的异常检测方法可行,能应对内部威胁。

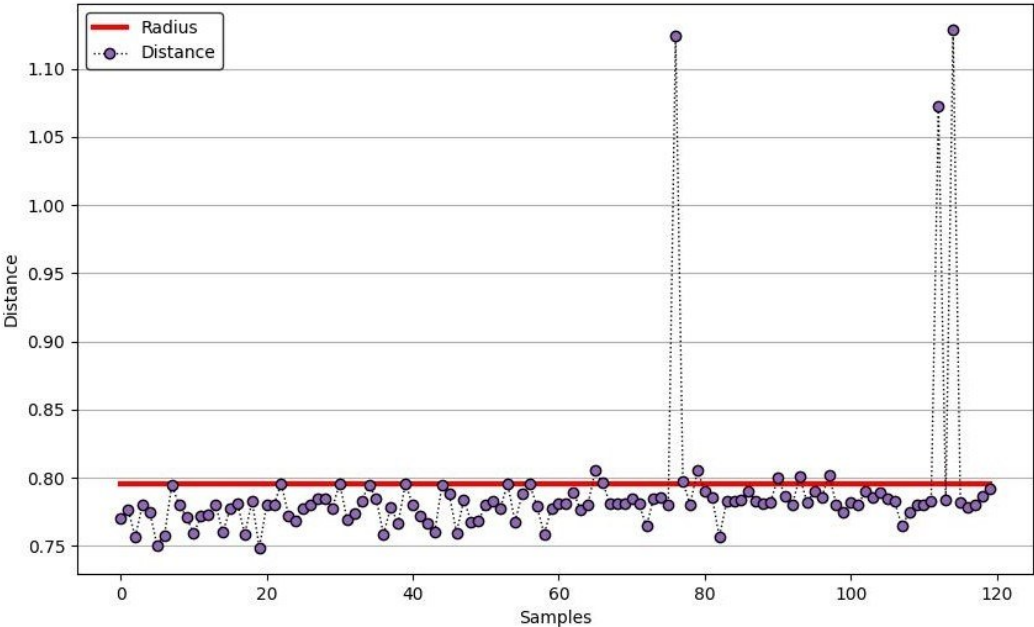


Fig. 11 Test results

图 11 检测结果

Table 2 Comparison of experimental results

表 2 实验结果比较

指标	本文结果	单类支持向量机	随机森林
查准率 P	0.93	0.94	0.92
查全率 R	0.91	0.86	0.91
F1 得分	0.92	0.89	0.88

5 结语

本文基于零信任理念提出一种 SDN 动态访问控制架构,采用属性签名验证进行用户身份认证,并在用户访问过程中持续进行身份验证。同时,基于信任度量进行用户动态访问控制,根据用户出现的异常行为动态调整访问权限。该架构为 SDN 提供了用户访问过程的安全管控,能应对 SDN 网络缺乏有效身份认证和内部威胁检测方法等问题,为实际 SDN 架构的部署提供了参考。

实验表明,架构实际部署的效果相较传统方法优势明显,但仍存在一些问题,需要将架构部署到实际环境中进行测试才能得以逐步解决。未来,将考虑更多用户的接入认证方式,综合研究更安全、快速的 SDN 认证方法;可加入用户组模型等多元模型,以多方位检测内部威胁。

参考文献:

[1] JIANG J F. Key technology analysis and development trend of SDN[J]. Software Guide, 2015 ,14 (6): 160-163.

[2] ELIYAN L F, DI PIETRO R. DoS and DDoS attacks in software defined networks: a survey of existing solutions and research challenges[J]. Future Generation Computer Systems, 2021, 122: 149-171.

[3] WABI A, IDRIS I, OLANI O M, et al. DDoS attack detection in SDN: method of attacks, detection techniques, challenges and research gaps [J]. Computers & Security, 2023, 139: 103652-103685.

[4] JARABA F, MAHAJAN G, JANI J, et al. Exploring current solutions against DDoS attacks in SDN environment [J]. Procedia Computer Science, 2024, 238: 127-134.

[5] GUO L, JING S, WEI L, et al. Defense method of crossfire attack based on software defined network [J]. Computer Engineering, 2024, 50(10): 216-227.

[6] ZHANG Q Q, LI Y Z. Research on SDN based DDoS attack detection technology[J]. Software Guide, 2019, 18 (7): 205-208.

[7] CHANG C W, JIN J S, HAN P S, et al. An SDN packet forwarding verification scheme based on attribute signature identification [J]. Journal of Communications, 2021, 42(6): 131-144.

[8] WANG S Y, LI Q, ZHANG Y. A lightweight SDN packet forwarding verification[J]. Chinese Journal of Computers, 2019, 42(1): 176-189.

[9] ALRASHE H A, ALHAMMAD F E, ALJHANI A M A, et al. A block-

- chain-based security framework for east-west interface of SDN[J]. *Electronics*, 2024, 13(19): 3799.
- [10] GAO N, PENG J, WANG S X. Research on zero trust security model [J]. *Journal of Cyber Security Research*, 2024, 10(10): 886–895.
高能, 彭佳, 王识潇. 零信任的安全模型研究[J]. *信息安全研究*, 2024, 10(10): 886–895.
- [11] OUIAZANE S, ADDOU M, BARRAMOU F. A zero-trust model for intrusion detection in drone networks [J]. *International Journal of Advanced Computer Science and Applications*, 2023, 14(11): 525–537.
- [12] DHIMAN P, SAINI N, GULZAR Y, et al. A review and comparative analysis of relevant approaches of zero trust network model [J]. *Sensors*, 2024, 24: 1328.
- [13] SALLAM A, REFAEY A, SHAMI A. On the security of SDN: a completed secure and scalable framework using the software-defined perimeter [J]. *IEEE Access*, 2019, 7: 146577–146587.
- [14] GUO X, XIAN H B, FENG T, et al. An intelligent zero trust secure framework for software defined networking [J]. *PeerJ Computer Science*, 2024, 9: e1674.
- [15] LIN Y F, CHEN X, XU Y Y, et al. A power sensitive data access control method based on the zero trust security model [J]. *Information Security Research*, 2024, 10(1): 88–93.
林奕夫, 陈雪, 许媛媛, 等. 基于零信任安全模型的电力敏感数据访问控制方法[J]. *信息安全研究*, 2024, 10(1): 88–93.
- [16] LI W X, ZHANG J H, ZENG J J, et al. A zero trust anonymous access scheme based on software-defined perimeter [J]. *Computer Science*, 2024, 51(12): 293–302.
李惟贤, 张建辉, 曾俊杰, 等. 基于软件定义边界的零信任匿名访问方案[J]. *计算机科学*, 2024, 51(12): 293–302.
- [17] XIAO J X, GUO Y B, CHANG C W, et al. Zero trust management of data flows between IoT edge nodes based on SDN [J]. *Journal of Communications*, 2024, 45(7): 101–116.
肖警续, 郭渊博, 常朝稳, 等. 基于SDN的物联网边缘节点间数据流零信任管理[J]. *通信学报*, 2024, 45(7): 101–116.
- [18] CHAKRAVARTHY V D, AMUTHA B. A critical analysis of Zero Trust Architecture (ZTA) [J]. *International Journal of Communication*, 2022, 35(2): 1–16.
- [19] LIU Y M, WANG Y, FENG H. A proactive defense method against eavesdropping attack in SDN-based storage environment [J]. *Cybersecurity*, 2024, 7: 58.
- [20] ZHANG Z S, HUANG W, CAI S Y, et al. An efficient reusable attribute-based signature scheme for mobile services with multi access policies in fog computing [J]. *Computer Communications*, 2022, 196: 9–22.
- [21] MISHRA R K, YADAV R K, NATH P. Access control models and frameworks for the IoT environment: review, challenges, and future direction [J]. *Wireless Personal Communications*, 2024, 138: 1671–1701.
- [22] ARTIOLI P, MACI A, MAGRÌ A. A comprehensive investigation of clustering algorithms for user and entity behavior analytics [J]. *Frontiers in Big Data*, 2024, 7: 1375818.
- [23] ZUBAIR M, SABZEVARI M, KHATRI V, et al. Access control for trusted data sharing [J]. *EURASIP Journal on Information Security*, 2024, 30: 1–30.
- [24] RUFF L, VANDERMEULEN R, GOERNITZ N, et al. Deep one-class classification [C]// *International Conference on Machine Learning*, 2018: 4393–4402.
- [25] SARHAN B, ALTWAIJRY N. Insider threat detection using machine learning approach [J]. *Applied Sciences*, 2023, 13(259): 1–19.

(责任编辑:刘嘉文)