

高级逃逸攻击检测技术研究综述

王英翔¹, 王 恒^{1,2,3}, 李雨琛⁴, 杨天宇¹

(1. 宁夏大学 信息工程学院; 2. 宁夏“东数西算”人工智能与信息安全重点实验室; 3. 宁夏大数据与人工智能
省部共建协同创新中心; 4. 国家能源集团 宁夏煤业有限责任公司煤制油分公司, 宁夏 银川 750021)

摘 要: 高级逃逸攻击是一种通过组合多种逃逸技术实现的新型攻击手段, 对现代网络防御体系构成了重大挑战。传统研究多聚焦于单一逃逸手段检测, 缺乏对跨层次高级逃逸攻击的系统性研究。系统阐述了高级逃逸攻击定义、攻击方式、攻击技术分类、常用数据集及检测方法, 重点对近5年高级逃逸攻击检测研究进展进行了梳理。对相关检测方法 & 数据集的分析表明, 高级逃逸攻击相关研究取得了一定进展, 但部分检测方法过度依赖于数据集, 对未知攻击的检测能力不足, 且现有公开数据集具有明显的标记特征, 可能导致模型在真实场景下泛化能力不足。最后给出了研究面临的挑战和展望。

关键词: 高级逃逸攻击; 检测方法; 跨层次攻击; 未知攻击; 数据集泛化

DOI: 10.11907/rjtk.251377

扫描二维码阅读全文:

中图分类号: TP393.08

文献标识码: A

文章编号: 1672-7800(2026)008-0199-10



A Review of Research on Advanced Evasion Attack Detection Techniques

WANG Yingxiang¹, WANG Heng^{1,2,3}, LI Yuchen⁴, YANG Tianyu¹

(1. School of Information Engineering, Ningxia University; 2. Ningxia Key Laboratory of Artificial Intelligence and Information Security for Channeling Computing Resources from the East to the West; 3. Ningxia Big Data and Artificial Intelligence Provincial-Ministerial Collaborative Innovation Center; 4. National Energy Group Ningxia Coal Coal-to-Oil Branch, Yinchuan 750021, China)

Abstract: Advanced evasion attack is a new type of attack achieved by combining multiple evasion techniques, which poses a major challenge to modern network defence system. Traditional research focuses on the detection of single evasion means, but lacks systematic research on cross-level advanced evasion attacks. In this paper, the definition of advanced evasion attacks, attack methods, classification of attack techniques, commonly used data sets and detection methods are systematically elaborated. It focuses on the research progress of advanced evasion attack detection in the past 5 years, and through the analysis of related detection methods and datasets, it shows that the research on advanced evasion attacks has made some progress, but some detection methods are overly dependent on datasets, and have insufficient detection ability for unknown attacks; the existing public datasets have obvious marking characteristics, which may lead to insufficient generalisation ability of the model in real scenarios. Finally, the challenges and outlook of the research are given.

Key Words: advanced evasion attack; detection method; cross-level attack; unknown attack; dataset generalisation

0 引言

高级逃逸攻击(Advanced Evasion Techniques, AET)相较于普通逃逸攻击, 通常具有更高的复杂性和隐蔽性。攻击者通过精心设计的攻击载荷和手段, 绕过防火墙、入侵

检测系统(IDS)和入侵防御系统(IPS)等传统安全设备, 使得攻击能够在不被察觉的情况下顺利执行。这种攻击方式的高效性和难以检测的特性, 使其成为现代网络安全防御中的一大难题。近年来, 发生了许多涉及高级逃逸攻击的安全事件。例如, 2022年攻击者利用Log4shell漏洞绕过日志记录系统的安全限制, 远程执行代码, 渗透了全球数

收稿日期: 2025-07-24

基金项目: 国家自然科学基金项目(62463025); 宁夏回族自治区重点研发计划项目(2022BDE03008)

作者简介: 王英翔(1998-), 男, 宁夏大学信息工程学院硕士研究生, 研究方向为流量识别、高级逃逸技术识别; 王恒(1975-), 男, 博士, 宁夏大学信息工程学院教授, 研究方向为网络空间安全、自然语言处理; 李雨琛(1999-), 男, 国家能源集团宁夏煤业有限责任公司煤制油分公司助理工程师, 研究方向为煤气化单元自动化控制; 杨天宇(2000-), 男, 宁夏大学信息工程学院硕士研究生, 研究方向为网络入侵检测。本文通讯作者: 王恒。

千家企业^[1]。同年,攻击者利用Spring4Shell中的数据绑定漏洞规避输入验证,绕过安全防护措施,远程执行代码^[2]。2023年,攻击者利用MOVEit Transfer软件中的SQL注入漏洞规避数据库的查询限制,绕过身份验证和访问控制,窃取敏感数据^[3]。此外,多起勒索病毒事件(如REvil、LockBit 3.0等)通过加密通信和伪装绕过企业的网络防护,受害者大多为全球大型公司,部分攻击者还在攻击后进行了大规模数据泄露^[4-5]。2023年,攻击者利用Follina(CVE-2022-30190)漏洞规避应用程序沙箱,绕过传统防护措施,执行任意代码^[6]。同年,攻击者通过篡改3CX Desktop App的软件更新,植入后门并绕过传统安全防御系统,影响了全球数十万用户^[7]。

本文主要工作如下:①对近5年的检测文献进行了梳理,从特征和模型两个维度对检测方法作出分类,并从攻击类型、特征分析、数据集、先验知识、实时检测、真实场景等方面对检测方法进行了比较;②对部分检测方案中的数据样本和数据集进行分析,发现部分数据集具有明显的标记特征,导致模型在真实场景中的泛化能力不足;③讨论了主动防御、被动防御机制和多层防御机制,分析了面临的挑战,并提出未来展望。

1 高级逃逸攻击概述

1.1 高级逃逸攻击的定义

高级逃逸攻击是一种通过组合多种逃逸技术(如代码混淆、协议滥用、内存注入等)并跨越多层防御机制(网络层、主机层、应用层)的新型攻击手段,攻击者通过运用复杂的策略,技术和工具攻击系统的弱点,使恶意活动在安全防护系统内保持隐蔽。这种高度隐蔽性和适应性的攻击方式使得网络安全防御面临更大的挑战。

1.2 高级逃逸攻击方式

高级逃逸攻击方式多种多样,其具体形式通常与攻击者的意图、攻击场景密切相关,主要包括以下4个方面:

(1)基于主机的攻击方式。基于主机的攻击方式是指攻击者在目标主机上利用恶意软件、文件注入、PowerShell等技术实施逃逸攻击,从而绕过主机级的安全防护措施,执行恶意操作,获取系统控制权或进行其他恶意行为。一些攻击者利用PowerShell和WMI等工具实施逃逸攻击,从而规避检测机制并进行信息窃取^[8-10]。

(2)基于网络通信协议的攻击方式。基于网络通信协议的攻击方式是指通过特定的协议或协议层(如DNS、HTTP、ICMP等)进行逃逸攻击。攻击者通常利用隧道技术,将恶意数据隐藏在正常的网络流量中,以绕过防火墙、入侵检测系统(IDS)和其他安全防护机制^[11-12]。

(3)基于文件漏洞的攻击方式。基于文件漏洞的攻击方式是指攻击者通过构造恶意文件,利用文件解析中的漏洞,规避安全检测机制。常见的文件逃逸载体包括文档文

件、图像文件、音频文件等。

1.3 高级逃逸攻击技术分类

1.3.1 载荷逃逸

载荷逃逸通常是指攻击者在渗透或攻击目标主机的过程中,首先通过改变恶意代码的形态或结构隐藏恶意行为,使其躲过传统的防病毒软件,IDS/IPS的检测进入目标主机中并执行恶意载荷。常用的技术包括多阶段攻击、代码注入、利用工具、文件分解和重组、反沙箱技术等。①多阶段攻击:攻击者分多个阶段释放恶意载荷,先通过邮件或网络漏洞获取一个较小的载荷,再在目标系统中下载并执行更复杂、更强大的攻击载荷^[11];②代码注入:恶意软件将代码注入到正常的应用程序或进程中,利用这些载体传播恶意载荷^[12];③利用工具:攻击者常使用操作系统自带的工具或第三方合法工具(PowerShell、WMI等)隐藏恶意载荷^[13];④文件分解和重组:恶意软件将恶意载荷分成多个小部分,并将其隐藏在不同文件中,待到达目标系统后再进行重组并执行^[14];⑤反沙箱技术:反沙箱技术是指一些恶意软件如果检测到自身处于沙箱环境中,则会中止自身运行或隐藏威胁,从而避免被发现^[15]。

1.3.2 协议逃逸

协议逃逸通过操控网络协议的方式,使得攻击流量可以绕过防火墙、IDS/IPS等网络安全工具。常用的技术包括利用协议构建通信隧道、协议滥用、协议实现混淆、协议分段等。①利用协议构建通信隧道:攻击者利用协议构建通信隧道(如VPN、DNS、HTTP等),将恶意流量隐藏在合法的网络流量中,使得恶意流量在通过防火墙或IDS时不会被识别出来^[16-18];②协议滥用:攻击者利用协议的一些特性和不常见的用法规避检测。例如,使用TCP连接中未定义或不常见的控制位,或利用协议中的冗余信息^[19];③协议实现混淆:攻击者通过对协议载荷进行非加密的形态变换,干扰基于规则匹配的安全检测(如使用使用Base64、URL编码等伪装恶意指令)^[20];④协议分段:攻击者将恶意数据或命令分割成多个小数据包,从而绕过网络防火墙或入侵检测系统(IDS)的检查^[21]。由于这些防护系统可能不会重组所有分段,因此无法识别此类恶意攻击。

1.3.3 流量逃逸

流量逃逸通过各种技术和手段伪装恶意流量使其看起来像正常流量,绕过网络安全设备(如防火墙、入侵检测系统(IDS)等)的检测与阻止。常用技术包括加密会话、端口绕过。①加密会话:攻击者通过加密技术(如SSL/TLS)隐藏真实数据内容,使得防火墙和IDS无法分析流量,这种技术通常被用于隐藏恶意命令、数据泄露或入侵行为^[22];②端口绕过:攻击者通过不同的技术或方法绕过网络中的端口控制或访问限制,达到绕过防火墙和IDS的目的^[23]。

1.3.4 多态与变种逃逸

多态与变种逃逸是指基于恶意载荷的自我变异和动态修改,使得攻击载荷每次表现出来的形态都不同,从而绕过基于签名的检测系统。常用的技术包括多态攻击、变种攻击。①多态攻击:攻击载荷在每次执行时通过加密和混淆技术改变代码的表现形式,但不改变原有的功能和行为,这使得反病毒软件和IDS无法通过签名检测到攻击^[24];②变种攻击:攻击者通过对恶意代码的源代码进行

修改或引入大量冗余操作、无意义指令、加密等手段改变代码的结构^[25-26]。这些修改会使得每个变种的恶意软件在行为上保持相同,但其具体实现存在显著差异。

1.4 目标与技术手段关联分析

高级逃逸攻击的攻击方式与技术手段之间存在着紧密关联,攻击者会根据其攻击目标和攻击场景选择对应的攻击方式和逃逸技术,它们之间的关联如表1所示。

Table 1 Correlation between attack methods and technical means

表1 攻击方式和技术手段之间的关联

攻击方式	相关逃逸技术	具体手段	应用场景
基于主机的攻击方式	载荷逃逸和协议逃逸	在主机上利用漏洞进行逃逸攻击	虚拟环境中的逃逸 Web应用中的逃逸
基于网络通信协议的攻击方式	协议逃逸和流量逃逸	使用隧道技术检测c&c信道	APT攻击,长期潜伏的恶意软件
基于文件漏洞的攻击方式	流量逃逸,协议逃逸	利用文件解析中的漏洞,规避检测机制	网络钓鱼邮件 文件传输

2 样本数据与数据集分析

2.1 样本数据

高级逃逸攻击检测对数据的丰富性和多样性提出了较高要求,研究者需要综合考虑不同的媒介和数据来源,以全面捕捉潜在的高级逃逸攻击行为。下文将介绍3种常用的样本数据类型。

2.1.1 网络流量数据

网络流量数据是检测流量逃逸、协议逃逸的关键数据来源之一,它基于数据包数据、流数据、会话数据和应用层数据进行划分。通过对网络流量数据的分析,可以识别出异常的流量模式。Huang等^[27]利用流级数据(如源IP地址、ACK标志、目的IP等)检测流量逃逸行为。Rosay等^[28]

分析应用层级数据的特征(如HTTPSPOST请求数和DNS查询频率等特征)实现协议逃逸的检测。此外,Pandit等^[29]通过提取会话级数据中的流持续时间、数据包总数等特征,使用HAGRU模型检测流量逃逸,表现出较高的检测精度。网络流量数据特征如图1所示。

2.1.2 系统行为数据

系统行为数据是检测多态与变种逃逸的数据来源之一,它包括文件操作行为数据、进程行为数据、注册表行为数据、网络行为数据和系统调用行为数据等。Zheng等^[29]将文件操作数据作为行为特征,成功检测多态与变种逃逸攻击。此外,Fu等^[30]利用进程行为数据(如加密时间、文件访问频率等)实现了对变种与多态逃逸攻击的有效检测。系统行为数据特征如图2所示。

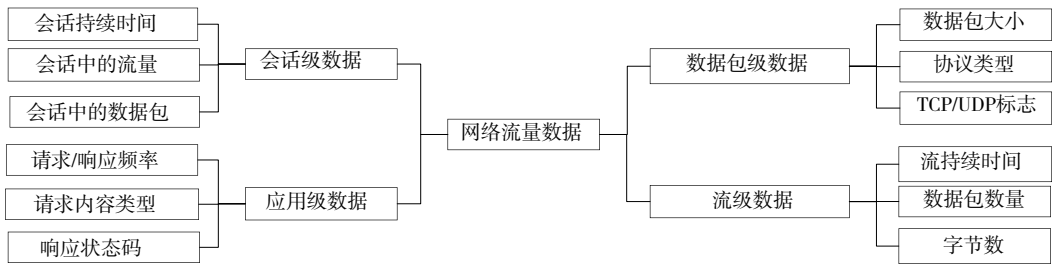


Fig. 1 Common features of network data

图1 网络数据常用特征

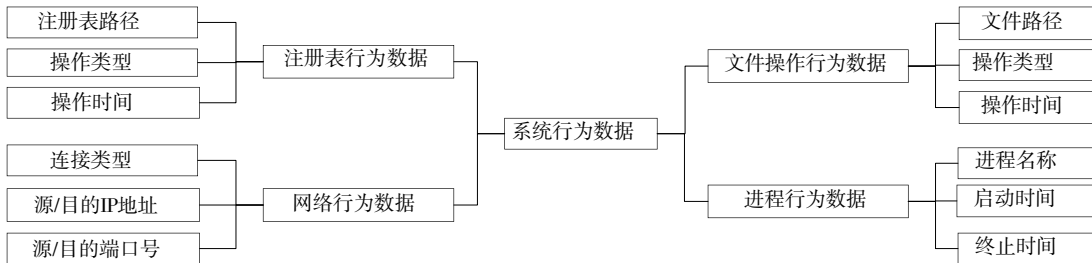


Fig. 2 Common features of system behaviour

图2 系统行为常用特征

2.1.3 日志数据

日志数据是检测载荷逃逸和多态与变种逃逸的数据来源之一,包括系统日志、网络设备日志、应用程序日志等。通过对日志数据的分析,可以发现攻击者在系统中的痕迹。近年来,一些研究者已经开始利用日志数据进行高级逃逸攻击检。Welderman 等^[31]从网络流量和系统日志

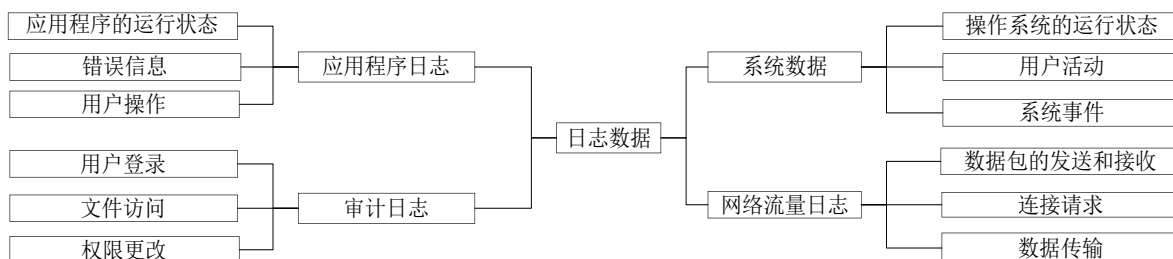


Fig. 3 Common features of log data

图3 日志数据常用特征

2.2 数据集

2.2.1 公开数据集

NSL-KDD 数据集作为经典的入侵检测基准数据集,包含 41 个特征维度,涵盖流量特征(如数据包大小、传输速率、连接数)、协议特征(如 TCP 标志、服务类型、错误率)以及统计特征(如连接持续时间、流量分布)等。该数据集共包含 148 000 条记录,涉及流量逃逸、多态与变种逃逸等多种攻击类型。然而,与其他数据集相比,NSL-KDD 存在数据重复、类别不平衡和时效性不足等局限性,因而更适合用于小规模研究。

CIC-AndMal2017 和 CIC-IDS2017 数据集由加拿大新不伦瑞克大学网络安全研究所通过专用网络流量生成器和攻击模拟工具构建。其中,CIC-AndMal2017 包含 4 352 个恶意软件样本和 6 500 个良性样本,而 CIC-IDS2017 则涵盖了更为丰富的攻击类型。这些数据集的特征集包括流量特征(如数据包大小、传输速率、流量方向)和行为特征(如会话持续时间、请求频率、错误率),并涉及流量逃逸、多态与变种逃逸等攻击类型。由于其攻击类型多样且覆盖范围广泛,该数据集被广泛应用于基于流量的逃逸攻击检测研究。例如,Loco 等^[33]和 Yu 等^[34]利用该数据集进行模型训练与性能评估,取得了显著的研究成果。

CIRA-CIC-DoHBrw-2020 数据集由加拿大网络安全创新中心通过 Dns2tcp、Dnscat2、Iodine 等隧道工具模拟攻击收集,包含 23 个特征维度和 1 000 000 条 DNS-over-HTTPS 流量记录,特征涵盖 DNS 查询、HTTPS 流量和时间序列等。该数据集专门用于检测协议逃逸行为,例如 Wang 等^[35]基于该数据集构建 1D-CNN 模型,在 DNS 协议逃逸检测中取得了较高的准确率。然而,与其他数据集相比,该数据集的数据类型相对单一,应用范围较为特定。

VirusShare 和 Malware Bazaar 数据集主要通过社区软件上传方式收集恶意软件样本及其行为数据,包含静态特征(如文件哈希、PE 头信息、字符串特征)、动态特征(如

中提取勒索软件的行为特征,实现了对高级逃逸攻击的检测。此外,Shanks 等^[32]通过分析系统日志数据(如 Windows 操作系统中的文件系统 I/O 请求包(IRP)操作等特征),成功检测载荷逃逸行为。日志数据特征如图 3 所示。

API 调用序列、网络行为、文件操作)以及多态特征(如代码混淆、加密、变种行为)。这些数据集包含大量真实世界的恶意软件样本,特别适用于多态与变种行为研究。例如,Zheng 等^[29]和 Fu 等^[30]利用该数据集进行了深入的特征分析,为恶意软件检测提供了重要参考。

MaleVis 数据集是通过在沙箱环境下运行恶意软件收集而来,包含 50 个特征维度,涵盖系统调用、网络行为和文件操作等。该数据集被广泛应用于半监督学习和无监督学习的恶意行为检测方法研究。例如,Albin 等^[36]利用该数据集进行载荷逃逸等恶意行为检测,取得了显著的研究成果。CTU-BOTNET 数据集由捷克技术大学通过模拟 BOTNET 攻击场景并利用专用工具收集,包含 20 多个特征维度,涵盖 BOTNET 通信、流量和协议特征。

2.2.2 自定义数据集

部分研究使用私有数据集开展实验。例如,Xu 等^[37]从网络安全公司的专有数据集以及遭受恶意软件攻击的企业提交的数据中收集样本,包含多个恶意软件变种,如 30 个 LockBit 2.0 样本、35 个 Conti 样本、25 个 Hive 样本、40 个 BlackMatter 样本、28 个 AvosLocker 样本、34 个 Revil (Sodinokibi)样本和 32 个 DarkSide 样本等。这类数据集常被用于检测多态与变种逃逸行为,为恶意软件研究提供了重要的数据支持。表 2 总结了各公共数据集和自定义数据集的收集方法、采集工具、数据类型和适用场景。

2.2.3 数据集分析

通过分析数据集发现,部分文献中未公开其所使用的数据集,导致其他研究者无法复现其实验结果。例如,一些文献中提到的专有数据集(如网络安全公司的恶意软件库)未公开,限制了研究的可重复性。并且,研究者在收集和使用数据集时出现了以下 3 个问题:

(1)数据实时性。部分数据集不实时更新,如 CIC-AndMal2017 数据集包含的数据是 2017 年实时收集的,难以反映当前逃逸攻击的最新动态。Liu 等^[38]在整合

Table 2 Common datasets for advanced evasion attacks
表 2 高级逃逸攻击常用数据集

数据集	时间	适用场景	收集方法	收集工具	数据类型
CIC-IDS2017	2017	检测协议逃逸和流量逃逸	通过模拟真实网络环境中的攻击场景以收集数据	Wireshark	正常流量和攻击流量,如 DoS、DDoS、Web 攻击等
CIC-AndMal2017	2017	检测多态与变种逃逸、流量逃逸	模拟 Android 恶意软件在移动设备上的运行行为	DroidBox	Android 恶意软件样本的行为数据
CIRA-CIC-DoHBrw-2020	2020	检测协议逃逸和流量逃逸	模拟 DNS 和 DoH 协议中的逃逸攻击	Wireshark	DNS 查询数据、DoH 流量数据、加密流量数据
VirusShare	2011	检测多态与变种逃逸	通过社区共享和恶意软件上传收集恶意软件样本	Python 脚本	恶意软件样本(包括多态攻击和变种攻击样本)
Malware Bazaar	2019	检测多态与变种逃逸	在沙箱环境中运行恶意软件,捕获其行为	Cuckoo Sandbox	恶意软件行为数据(如文件访问模式、网络交互)
MaleVis	2019	检测载荷逃逸	通过社区共享和恶意软件上传收集恶意软件样本	Python 脚本	恶意软件样本(包括多态攻击和变种攻击样本)
NSL-KDD	2009	检测流量逃逸	通过网络流量分析工具收集	tcpdump	正常流量和攻击流量
CTU-BOTNET	2011	检测流量逃逸和协议逃逸	模拟 Botnet 攻击场景	tcpdump	Botnet 攻击流量数据
自定义数据集	2024	检测多态与变种逃逸	受攻击单位私有真实数据集	DroidBox tcpdump	LockBit 2.0 样本、Conti 样本、Hive 样本等

多个数据集进行研究时,也间接反映了单一老旧数据集在应对新型威胁时的局限性。针对该问题,可通过数据增强技术加以解决,如对系统行为数据进行特征扰动和对恶意样本进行可控变异(在保持核心恶意功能的前提下,修改其逃逸技术实现方式),生成新变种。

(2)数据多样性。如 CIC-AndMal2017 数据集主要关注 Android 恶意软件且样本主要来自 Google Play 应用商店,无法代表所有恶意软件变种;DoHBrw-2020 专注于 DOH 协议的逃逸攻击,如 Gurierrez 等^[39]、Chen 等^[40]的研究所示,该数据集缺乏对 HTTP/2、QUIC 等新兴协议或其他类型逃逸的覆盖(如基于文件漏洞的逃逸)。针对该问题,可通过实时数据收集加以解决,如部署高交互蜜罐系统(High-interaction Honeypot Systems)主动诱捕和分析最新攻击。

(3)数据标注问题。如 CIC-AndMal2017 中的恶意软件具有多种行为特征,但仅标注为单一恶意类别,而未区分具体使用的逃逸技术(如代码注入、协议隧道)等。这种粗粒度的标注导致模型无法学习不同逃逸技术的差异特征,降低了检测混合逃逸攻击的精度,这可能是导致 Wang 等^[35]、Fu^[30]等研究在检测多态与变种逃逸时,模型对技术细节区分度不够的原因之一。针对该问题,可以细粒度标注加以解决,如在数据集构建过程中,采用多标签标注(Multi-Label Annotation)或结构化元数据,详细记录样本所使用的逃逸技术类别(如载荷逃逸—代码注入、协议逃逸—DNS 隧道、流量逃逸—端口滥用等)以及攻击的阶段、目标平台等信息。

分析还发现,研究者公开的数据存在参差不齐的状况。如 CIC-IDS2017 数据集中存在大量重复数据,尤其是正常流量数据重复比例较高,这会导致模型出现过拟合现象,无法反映出模型性能。

3 高级逃逸攻击检测方法

本文从模型和特征两个维度对现有的高级逃逸攻击检测方法进行分类,从特征维度可分为基于流特征和基于行为特征的检测方法,从模型维度可分为基于机器学习和基于深度学习的检测方法。

3.1 基于流特征的检测方法

基于流特征的检测方法主要基于网络通信流量的统计特征和通信模式展开分析,通过识别通信流量的异常模式以发现潜在攻击。流特征通常来源于数据包数据、流数据、应用层数据和会话层数据。

有研究者从流数据中提取特征以进行模型训练。Huang 等^[27]通过分析 IoT 设备中的流量特征,包括持续时间、ack 标志、目的 IP 等,并结合递归神经网络(Recurrent Neural Network, RNN)和卷积神经网络(Convolutional Neural Networks, CNN)模型检测流量逃逸,准确率和召回率达到了 99%。Rosay 等^[28]将流特征和拓扑特征融合,并使用图卷积神经网络(Graph Convolutional Networks, GCN)检测流量逃逸。

为了应对更复杂的协议逃逸和流量逃逸,研究者从应用层数据和会话层数据中提取特征。Pandit 等^[41]利用 HTTP POST 请求数量和 DNS 服务器查询频率等特征,结合随机森林检测协议逃逸。Meng 等^[42]通过分析 TCP/UDP 数据包,提取了 13 个特征,包括(ACK 标志和生存时间(TTL)特征),使用深度神经网络检测流量逃逸。Udiyon 等^[43]从数据包报头中提取了 29 个连接特征(单个连接的平均持续时间和连接数),并训练神经网络检测流量逃逸。Van 等^[44]提取数据流中的流持续时间、数据包长度、数据包速率、连接状态等特征,并使用结合层次化注意力机制的深度神经网络模型检测流量逃逸。

为了提高检测方法的时效性和适用性,Alauthama

等^[45]提出一种深层次网络模型,并直接从原始数据包中学习特征,通过一维卷积和GRU提取时空特征,显著提高了检测时效性。Liu等^[38]提出一种加密流量逃逸检测方法,结合合成数据增强技术和基于粒子群优化器的多样化集成模型,并整合多个数据集(inSD、UNSWNB、ioTID-20)用于训练和测试。Wang等^[35]提出一种名为TLARNN的方法,其结合了一维卷积神经网络1D-CNN和双向门控循环单元BiGRU,从数据包层中提取流特征,检测流量逃逸。实验结果表明,该方法在性能和时效性上均优于文献[38]

中提出的方法。

此外,Rustam等^[46]提出一种名为GCN-ETA的加密恶意流量检测方法,通过改进的GCN提取流量统计特征和轨迹图的特征,并使用决策树检测流量逃逸。Liu等^[47]提出Whisper系统,将数据包序列编码为向量并进行离散傅里叶变换(Discrete Fourier Transform),利用轻量级聚类算法对特征进行聚类,从而检测流量逃逸。表3从方法、攻击类型、特征分析、数据集、先验知识、实时检测、真实场景等方面对上述代表性研究作了详细总结。

Table 3 Summary of detection work based on flow features

表3 基于流特征的检测工作总结

文献	检测方法	攻击类型	特征分析	数据集	先验知识	真实场景	实时检测
Huang等 ^[27]	RNN+CNN	流量逃逸	静	IOT-23	依赖	×	√
Rosay等 ^[28]	RF	协议逃逸	动	自定义	依赖	√	√
van Roosmalen等 ^[44]	BIGRU+层次注意力机制	流量逃逸	静	NSLL-KDD	依赖	×	×
Alauthaman等 ^[45]	CNN+GRU	流量逃逸	动+静	CICIDS2017	依赖	×	√
Wang等 ^[35]	1D-CNN+BIGRU	流量逃逸	动	CICAndMal2017	依赖	×	×
Rustam等 ^[46]	SVM、RF	流量逃逸	动	Tianqiong Sandbox	依赖	√	×

注:“静”表示使用静态特征进行分析;“动”表示使用动态特征进行分析;√表示该方法可以适应该场景;×表示该方法不能适应该场景。

3.2 基于行为特征的检测方法

行为特征反映了用户或攻击者在系统或网络中的操作模式,能够有效区分正常与异常行为。由于恶意软件常通过高级逃逸攻击隐藏其真实意图,因此分析行为特征可以有效地发现攻击行为。有研究者通过分析系统行为检测高级逃逸攻击。Zheng等^[29]通过持续监控系统事件,提取行为特征(如文件访问模式、系统调用频率),并利用聚类算法构建行为签名,检测多态与变种逃逸及流量逃逸。Fu等^[30]提出基于时间行为分析的检测框架,通过分析时间行为特征(如加密时间、文件访问频率、系统修改时间间隔、网络活动序列等)检测多态与变种逃逸。Anikolova等^[48]提取系统级的加密行为特征,并通过数学建模和机器学习分析这些特征,以检测多态与变种逃逸。

随着多态与变种逃逸技术的快速发展,有研究者提出从系统行为数据和日志数据中提取行为特征。Welderman

等^[31]提出基于动态行为分析恶意软件检测框架,从网络流量和系统日志中提取勒索软件的行为特征,生成流签名并使用机器学习模型进行实时分析,且无需依赖预定义的签名。Shanks等^[32]通过监控系统中的文件变化提取行为特征,并结合机器学习检测载荷逃逸和流量逃逸,精度达到了95%。Loco等^[33]提出一种名为DELP的新型恶意软件检测方法,提取多层文件系统熵(Entropy)变化作为行为特征,结合实时熵分析,能够准确地识别多态与变种逃逸。

此外,Zhang等^[34]提出一种名为DBTP的自动化恶意软件检测方法,通过持续监控系统级事件(如文件I/O操作、网络异常和进程操作)构建行为特征,使用有监督算法检测流量逃逸和载荷逃逸,并支持动态更新检测模型以提高准确性。表4对上述代表性研究作了详细总结。

Table 4 Summary of detection work based on behavioural traits

表4 基于行为特征的检测工作总结

文献	检测方法	攻击类型	特征分析	数据集	先验知识	真实场景	实时检测
Zheng等 ^[29]	RF、SVM	多态与变种逃逸和流量逃逸	动+静	VirusShare CICAndMal2017	依赖	×	√
Anikolova等 ^[48]	K均值+Z分数分析	多态与变种逃逸	动	Malware Bazaar	依赖	×	√
Welderman等 ^[31]	RF、DT	载荷逃逸与流量逃逸	动	VirusShare CICAndMal2017	依赖	√	√
Shanks等 ^[32]	LSTM、GRU	多态与变种逃逸	动	VirusShare CICAndMal2017	依赖	×	√
Loco等 ^[33]	多层文件系统结构的熵值分析	载荷逃逸	动	VirusShare CICAndMal2017	不依赖	√	√
Zhang等 ^[34]	SVM、DT	流量逃逸,载荷逃逸	动+静	Malware Bazaar CICAndMal2017	极少依赖	×	√

注:“静”表示使用静态特征进行分析;“动”表示使用动态特征进行分析;√表示该方法可以适应该场景;×表示该方法不能适应该场景。

3.3 基于机器学习的检测方法

有研究者从系统行为数据和网络流量数据中提取特征,并将其作为机器学习模型的输入。Tarness等^[49]从系

统行为数据中提取特征,并使用多标签支持向量机(Support Vector Machines, SVM)进行分类,该方法不仅能够检测使用DNS协议进行逃逸攻击的恶意软件,还能识别多种

DNS 隧道类型(如 FTP-DNS、HTTP-DNS、HTTPS-DNS 和 POP3-DNS 隧道)。Limer 等^[50]从系统行为数据中提取特征,并通过数据可视化 and 对抗性训练,成功检测了多态与变种逃逸。Liu 等^[51]从网络流量中提取特征,并使用决策树算法构建二元分类模型,以检测流量逃逸。

部分研究者通过对比不同分类器的性能(如 F1 分数、精确度、召回率等指标)确定最佳分类器。Almusawi 等^[52]从文件操作系统中提取特征,并利用随机森林、支持向量机和梯度提升算法检测多态与变种逃逸,实验结果表明支持向量机算法效果最佳。Albin 等^[36]分别使用随机森林和支持向量机检测载荷逃逸,随机森林的表现优于支持向量机。

随着高级逃逸攻击手段的进步,依靠标签和先验知识的模型无法检测新型攻击,因此研究者开始关注无监督学习。Forsyth 等^[53]提出一种四层混合框架 RDS:第一层采用基于决策树检测已知的恶意行为;第二层通过堆叠集成模型和超参数优化技术提升性能;第三层利用 K-means 算法

进行无监督学习,检测未知的恶意行为;第四层采用超参数优化技术进一步优化无监督学习模型。Colangelo 等^[54]提出基于生成对抗网络(Generative Adversarial Networks, GAN)的检测方法,通过分析 Windows 操作系统中的文件系统 I/O 请求包(IRP)操作以识别多态与变种逃逸。

此外,Dib 等^[55]提出了基于双层随机森林的检测方法,通过分层分析操作码序列检测多态与变种逃逸。第一层采用广泛特征集进行训练,捕捉操作码分布的一般模式;第二层则专注于第一层识别出的重要特征。Xu 等^[37]通过分析字节码(Bytecode)的结构特征,并结合余弦相似度和机器学习分析特征,能够有效检测多态与变种逃逸。Moritaka 等^[56]提出基于聚合随机森林技术的检测方法,通过整合多个随机森林模型,能够有效检测包括使用高级逃逸技术的勒索软件变体在内的多种勒索软件。表 5 从方法、攻击类型、特征分析、数据集、先验知识、实时检测、真实场景以及鲁棒性等方面对上述代表性研究作了详细总结。

Table 5 Summary of detection work based on machine learning
表 5 基于机器学习的检测工作总结

文献	检测方法	攻击类型	特征	特征分析	数据集	先验知识	真实场景	实时检测	鲁棒性
Tarness 等 ^[49]	对抗性训练和机器学习	多态与变种逃逸	系统调用等	动	MSBIG 恶意软件数据库	部分依赖	×	×	H
Limer 等 ^[50]	核支持向量机(Kernel SVM)	协议逃逸	DNS 请求长度等	动	CIRA-CIC DoHBrw-2020	依赖	√	×	L
Liu 等 ^[51]	DT、SVM、FNN	流量逃逸	流特征	动	Android Ransomware Detection	依赖	×	√	H
Almusawi 等 ^[52]	CNN	多态与变种逃逸	操作码	动+静	自定义	依赖	×	√	H
Albin 等 ^[36]	半监督机器学习	载荷逃逸	操作码	动+静	MaleVis	部分依赖	×	×	L
Forsyth 等 ^[53]	DT、K-mean、堆叠集成模型	载荷逃逸	文件访问模式	动	BitcoinHeist	不依赖	×	×	H
Colangelo 等 ^[54]	GAN	载荷逃逸	I/O 请求包(IRP)操作	动	自定义	不依赖	√	×	M
Dib 等 ^[55]	双层随机森林	载荷逃逸	操作码序列等	动	自定义	依赖	×	×	M
Xu 等 ^[37]	余弦相似度+机器学习	多态与变种逃逸	字节码,操作码序列等	动+静	自定义	依赖	×	√	M
Moritaka 等 ^[56]	聚合随机森林	多态与变种逃逸	文件系统交互等	静	自定义	依赖	√	×	M

注:“静”表示使用静态特征进行分析;“动”表示使用动态特征进行分析;L(low)表示鲁棒性较差;M(middle)表示鲁棒性中等或具有一定的迁移性;H(high)表示鲁棒性好;√表示该方法可以适应该场景;×表示该方法不能适应该场景。

3.4 基于深度学习的检测方法

与传统的机器学习方法相比,深度学习模型能够直接从原始数据中学习特征,从而最大限度地保留信息并提升检测性能。近年来,研究者提出了多种基于深度学习的检测方法。Argene 等^[57]提出一种融合代码动态特征和静态特征的方法,结合卷积神经网络(CNN)和递归神经网络(RNN)学习恶意行为特征,检测精度达到 98%。Rafapa 等^[58]提出一种结合集成学习和迁移学习的恶意软件分类系统,即使面对多态与变种逃逸技术,分类准确率仍能达到 99.64%。

针对协议逃逸攻击,Gurierrez 等^[39]提出一种基于深度学习的检测方法,构建了一个由嵌入层、一维卷积层和密集连接层组成的 1D-CNN 模型,用于识别利用 DNS 协议进行逃逸的恶意行为。Chen 等^[40]提出一种端到端的异常检测模型,结合变分自编码器(Variational Auto-Encoder,

VAE)和自注意力机制,自动从原始流量数据中学习特征表示,并通过重构误差检测检测基于 DNS-over-HTTPS(DoH)协议的逃逸行为。

此外,Palau 等^[59]提出名为 MFC-DOH 的方法,用于在少量样本(Few-Shot)情况下检测基于 DNS-over-HTTPS(DoH)协议的逃逸行为。该方法结合了模型无关的元学习(MAML)和一种特殊的卷积神经网络(F-CNN),并引入了频域层和多头自注意力层(MHSA),显著提升了检测性能。此外,Ding 等^[60]提出一种基于深度神经网络(DNN)和关联分析的恶意流量检测系统,结合 DNN-4 和 Apriori 算法挖掘不同离散特征与正常标签之间的关联规则,以过滤分类后的流量。Liu 等^[61]利用深度强化学习(Deep Reinforcement Learning, DRL)增强恶意软件检测器的鲁棒性,能够有效检测使用高级逃逸攻击技术的恶意软件变体。表 6 对上述代表性研究作了详细总结。

Table 6 Summary of detection work based on deep learning
表 6 基于深度学习的检测工作总结

文献	检测方法	攻击类型	模型的输入	特征分析	数据集	先验知识	真实场景	实时检测	鲁棒性
Rafapa 等 ^[58]	CNN+RNN	多态与变种逃逸	代码签名,系统调用	动+静	Drebin	依赖	×	√	M
					VirusShare				
Gurierrez 等 ^[39]	1D-CNN	协议逃逸	域名长度等	动	CIRA-CIC-DoHBrw-2020	依赖	√	√	H
Chen 等 ^[40]	VAE+自注意力机制	协议逃逸	包长度,包方向	动	CIRA-CIC-DoHBrw-2020	部分依赖	√	√	H
Palau 等 ^[59]	MHSA+fcNN+MAML	协议逃逸	流字节数等	动	CIRA-CIC-DoHBrw-2020	依赖	√	×	H
Ding 等 ^[60]	DNN+Aprior算法	流量逃逸	数据包字节数等	动	NSL-KDD CIC-IDS2017	依赖	√	×	H
Liu 等 ^[61]	2DQN	流量逃逸	流量方向等	动	CTU BOTNET	依赖	√	×	H

注:“静”表示使用静态特征进行分析;“动”表示使用动态特征进行分析;L(low)表示鲁棒性较差;M(middle)表示鲁棒性中等或具有一定的迁移性;H(high)表示鲁棒性好;√表示该方法可以适应该场景;×表示该方法不能适应该场景。

3.5 检测方法面临的挑战与局限性

尽管许多检测方法在特定数据集上达到了高精度,但其在真实世界部署(Real-world Deployment),在跨数据集测试(Cross-dataset Evaluation)和面对对抗性攻击时,其性能往往表现出显著下降。

Rustam 等^[46]提出的 PSO-D-SEM 模型虽然在整合数据集(inSD、UNSW-NB15、IoTID-20)上表现良好,但其模型在单一数据集上训练后迁移到其他数据集时性能存在波动,F1 分数下降超过 10%,这凸显了模型对数据源特征的依赖性和泛化挑战。

Liu 等^[61]利用深度强化学习(DRL)生成对抗性恶意流量,成功使基于传统特征的 Botnet 检测模型的检测率从 90% 降至 70%,显著证明了模型在面对逃避性对抗攻击(Evasive Adversarial Attacks)时的脆弱性。

表 5 和表 6 对部分方法鲁棒性的标注(L/M/H)直观反映了其脆弱性。例如,Colangelo^[54]使用半监督学习在 Mal-eVis 数据集上检测载荷逃逸,其鲁棒性被标注为 L(Low),表明该方法对未知变种或轻微扰动的抵抗力较弱。Gurierrez 等^[39]结合 CNN 和 RNN 的方法虽然精度高(98%),但其鲁棒性标注为 M(Middle),提示其在面对针对性逃逸技术时可能存在漏洞。

4 结语

本文系统综述了高级逃逸攻击的检测和防御研究进展,分析了载荷逃逸、协议逃逸、流量逃逸、变种与多状态逃逸的特点与机制。梳理了研究中的检测方法,包括基于流特征、基于行为特征、基于机器学习和基于深度学习的检测技术,并进行了对比分析。这些检测方法在检测精度

上明显优于传统方法,但仍面临小样本场景下泛化能力不足和模型鲁棒性较差等挑战。

未来研究包括:①小样本学习与数据增强,开发新算法以提高检测模型在有限数据条件下的性能和泛化能力;②模型鲁棒性,通过对抗训练、迁移学习等技术增强模型对新型攻击和动态变化的适应性与抗攻击能力,旨在增强对未知攻击的检测。

参考文献:

[1] HIESGEN R, NAWROCK M, SCHMIDT T C, et al. The log4j incident: a comprehensive measurement study of a critical vulnerability [J]. IEEE Transactions on Network and Service Management, 2024, 21: 5921–5934.

[2] YAN X, HE B, SHEN W, et al. Automated data binding vulnerability detection for Java web frameworks via nested property graph [C]//Proceedings of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis, 2024: 1377–1388.

[3] KHANG T Y, RAHMAN N A A. Botnet as a service towards national defense and security [C]//Proceedings of the 14th International Conference on Developments in eSystems Engineering (DeSE), 2021: 86–91.

[4] RAUF ALI KHAN M. Understanding impacts of a ransomware on medical and health facilities by utilizing LockBit as a case study [J]. Security and Privacy, 2024, 7(1): e328.

[5] SINGH R, SINGH S, SINGH A. REvil ransomware [C]//Nagpur: The International Conference on Innovations and Challenges in Emerging Technologies, 2024.

[6] MOHAMED N. A comprehensive framework for cyber threat detection: leveraging AI, NLP, and malware analysis [J]. International Journal of Information Technology, 2025, 4: 1–8.

[7] FRANKRIG E. Automated binary differential analysis and behavior identification for closed-source software supply chain attack detection [D]. Nantredraeg County: Norwegian University of Science and Technology,

- 2024.
- [8] VARLIOHLU S. Detecting PowerShell-based fileless cryptojacking attacks using machine learning[D]. Cincinnati: University of Cincinnati, 2025.
- [9] CAO Y B. Detection of malicious PowerShell based on multi-label classifier[J]. Computer Knowledge and Technology, 2023, 19(19): 20–22.
曹耀彬. 基于多标签分类器的恶意PowerShell检测[J]. 电脑知识与技术, 2023, 19(19): 20–22.
- [10] VARLIOGLU S, ELSAYED N, VARLIOGLU E R, et al. The pulse of fileless cryptojacking attacks: malicious PowerShell scripts [C]//Atlanta: IEEE SoutheastCon 2024, 2024.
- [11] ZHENG Z Q, WANG R Q, FAN Z J, et al. A review of research on DNS tunnel detection technology [J]. Computer Applications, 2025, 45(7): 2079–2091.
郑智强, 王锐棋, 范子静, 等. DNS隧道检测技术研究综述[J]. 计算机应用, 2025, 45(7): 2079–2091.
- [12] AMIROV N, ISIK B, TUNCER B T, et al. DNS tunneling: threat landscape and improved detection solutions [DB/OL]. <https://arxiv.org/abs/2507.10267>.
- [13] HOMOU D L, BARPANDA N, BOBATO V, et al. Analysing a multi-stage cyber threat and its impact on the power system [J]. IET Cyber-Physical Systems: Theory & Applications, 2025, 10(1): e12107.
- [14] BAKIR R. UniEmbed: a novel approach to detect XSS and SQL injection attacks leveraging multiple feature fusion with machine learning techniques [J]. Arabian Journal for Science and Engineering, 2025, 50: 15591–15604.
- [15] ROSE A J, GRAHAM S R, SCHUBERT K C M, et al. ScriptBlock smuggling: uncovering stealthy evasion techniques in PowerShell and .NET environments [J]. Journal of Cybersecurity and Privacy, 2024, 4(2): 153–166.
- [16] DING Y, ZHANG M, LI X, et al. Defending against malicious code: a comprehensive study of evasion and mitigation techniques [C]//Proceedings of the International Conference on Computational & Experimental Engineering and Sciences, 2024: 376–389.
- [17] PANDEY B, SHUKLA M M, PANDEY P, et al. Development of malware in Windows sandbox and Kali MSFvenom for hacking Android and Windows operating systems [J]. International Journal of Information Technology, 2025, 17(3): 1831–1838.
- [18] SHARMA N, SWARNKAR M. DLAZE: detecting DNS tunnels using lightweight and accurate method for zero-day exploits [J]. IEEE Transactions on Network and Service Management, 2025, 22(3): 2343–2353.
- [19] ZUO M, GUO C, XU H, et al. METC: a hybrid deep learning framework for cross-network encrypted DNS over HTTPS traffic detection and tunnel identification [J]. Information Fusion, 2025, 121: 103125.
- [20] NESS S. Securing networks against adversarial domain name system tunneling attacks using hybrid neural networks [J]. IEEE Access, 2025, 13: 46697–46709.
- [21] OWOKO W. Survey on evolving threats in TCP/IP header attacks: emerging trends and future directions [J]. World Journal of Advanced Engineering Technology and Sciences, 2024, 11(2): 454–475.
- [22] ASGHAR H J, ZHAO B Z H, IKRAM M, et al. Use of cryptography in malware obfuscation [J]. Journal of Computer Virology and Hacking Techniques, 2024, 20(1): 135–152.
- [23] CHEN H. Enhancing the security of transmission control protocol (TCP): challenges and solutions for modern network threats [J]. Applied and Computational Engineering, 2025, 133: 46–53.
- [24] AHN J, HUSSAIN R, KANG K, et al. Exploring encryption algorithms and network protocols: a comprehensive survey of threats and vulnerabilities [J]. IEEE Communications Surveys & Tutorials, 2025, 27(6): 3587–3614.
- [25] EMMANUEL O I, AYODELE A A, ADEBIYI A M, et al. Windows firewall bypassing techniques: an overview of HTTP tunneling and nmap evasion [C]//Proceedings of the International Conference on Computational Science and Its Applications, 2021: 546–556.
- [26] MAURI L, DAMIANI E. Hardening behavioral classifiers against polymorphic malware: an ensemble approach based on minority report [J]. Information Sciences, 2025, 689: 121499.
- [27] HUANG G C, CHANG K C, LAI T H. Chaotic-based shellcode encryption: a new strategy for bypassing antivirus mechanisms [J]. Symmetry, 2024, 16(11): 1526.
- [28] ROSAY A, CARLIER F, CHEVAL E, et al. From CIC-IDS2017 to LYCOS-IDS2017: a corrected dataset for better performance [C]//Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2021: 570–575.
- [29] ZHENG J, ZENG Z, FENG T. GCN-ETA: high-efficiency encrypted malicious traffic detection [J]. Security and Communication Networks, 2022(1): 4274139.
- [30] FU C, LI Q, SHEN M, et al. Frequency domain feature based robust malicious traffic detection [J]. IEEE/ACM Transactions on Networking, 2022, 31(1): 452–467.
- [31] WELDERMAN G, CASTELLANOS R, WHITACRE A, et al. A robust system for ransomware detection using temporal behavior modeling [EB/OL]. <https://www.techrxiv.org/users/850300/articles/1237005/master/file/data/e391/e391>.
- [32] SHANKS G, STERLING M, HARRINGTON N, et al. Innovative framework for ransomware detection using adaptive cryptographic behavior analysis [EB/OL]. <https://www.authorea.com/doi/full/10.22541/au.173230401.11413813>.
- [33] LOCO P, ALONSO S, HARTMANN G, et al. Adaptive behavior-based ransomware detection via dynamic flow signatures [EB/OL]. <https://www.researchsquare.com/article/rs-5317374/v1>.
- [34] ZHANG H, ZHAO L, YU A, et al. Ranker: early ransomware detection through kernel-level behavioral analysis [J]. IEEE Transactions on Information Forensics and Security, 2024, 19: 6113–6127.
- [35] WANG B, SU Y, ZHANG M, et al. A deep hierarchical network for packet-level malicious traffic detection [J]. IEEE Access, 2020, 8: 201728–201740.
- [36] ALBIN A A, SHAAHID A, ALNASSER F, et al. Android ransomware detection using supervised machine learning techniques based on traffic analysis [J]. Sensors, 2023, 24(1): 189.
- [37] XU B, WANG S. Examining windows file system IRP operations with machine learning for ransomware detection [EB/OL]. <https://www.researchsquare.com/article/rs-4032456/v1>.
- [38] LIU X, LIU J. Malicious traffic detection combined deep neural network with hierarchical attention mechanism [J]. Scientific Reports, 2021, 11

- (1):12363.
- [39] GURIERREZ R, VILLEGAS-CH W, GODOY L N, et al. Application of deep learning models for real-time automatic malware detection [J]. *IEEE Access*, 2024, 12: 107742–107756.
- [40] CHEN Z, CAO J. VMCTE: visualization-based malware classification using transfer and ensemble learning [J]. *Computers, Materials & Continua*, 2023, 75(2): 4445–4465.
- [41] PANDIT A V, MONDAL D. Real-time malware detection on IoT devices using behavior-based analysis and neural networks [J]. *Research Journal of Computer Systems and Engineering*, 2023, 4(2): 117–129.
- [42] MENG X, LANG B, YAN Y, et al. Deeply fused flow and topology features for botnet detection based on a pretrained GCN [J]. *Computer Communications*, 2025, 233: 108084.
- [43] UDIYONO, AGUNG, LIM C, et al. Botnet detection using DNS and HTTP traffic analysis [C]//*Proceedings of the 2021 International Conference on Engineering and Information Technology for Sustainable Industry*, 2020: 1–6.
- [44] VAN ROOSMALEN J, VRANKEN H, VAN EEKELEN M. Applying deep learning on packet flows for botnet detection [C]//*Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, 2018: 1629–1636.
- [45] ALAUTHAMAN M, ASLAM N, ZHANG L, et al. A P2P botnet detection scheme based on decision tree and adaptive multilayer neural networks [J]. *Neural Computing and Applications*, 2018, 29: 991–1004.
- [46] RUSTAM F, JURCUT A D. Malicious traffic detection in multi-environment networks using novel S-DATE and PSO-D-SEM approaches [J]. *Computers & Security*, 2024, 136: 103564.
- [47] LIU J, WANG L, HU W, et al. Spatial-temporal feature with dual-attention mechanism for encrypted malicious traffic detection [J]. *Security and Communication Networks*, 2023(1): 7117863.
- [48] ANIKOLOVA E, MARTINS S, ROZENTAL D, et al. Ransomware detection through behavioral attack signatures evaluation: a novel machine learning framework for improved accuracy and robustness [DB/OL]. <https://www.techrxiv.org/users/849290/articles/1236401>.
- [49] TARNESS S, BENNETT M, HALLOWAY F, et al. Introducing dynamic entropy layer profiling: a novel approach for ransomware detection through behavioral feature analysis [EB/OL]. <https://www.research-square.com/article/rs-5358022/v1>.
- [50] LIMER A, ABRAMOVICH R, DEVEREUX G, et al. Automated ransomware detection using dynamic behavior trace profiling [DB/OL]. <https://www.techrxiv.org/users/847492/articles/1235305>.
- [51] LIU X, LIN Y, LI H, et al. A novel method for malware detection on ML-based visualization technique [J]. *Computers & Security*, 2020, 89(1): 101682.
- [52] ALMUSAWI A, AMINTOOSI H. DNS Tunneling detection method based on multilabel support vector machine [J]. *Security and Communication Networks*, 2018(1): 6137098.
- [53] FORSYTH G. Opcodes to images: a framework for early detection of ransomware by utilising machine learning techniques [C]//*Proceedings of the Wellington Faculty of Engineering Symposium*, 2023: 489.
- [54] COLANGELO L M. Malware family classification with semi-supervised learning [D]. Città di Torino: Politecnico di Torino, 2023.
- [55] DIB O, NAN Z, LIU J. Machine learning-based ransomware classification of bitcoin transactions [J]. *Journal of King Saud University-Computer and Information Sciences*, 2024, 36(1): 101925.
- [56] MORITAKA H, KOMURO D. Enhanced ransomware detection using dual-layer random forest on opcode sequences [EB/OL]. <https://ui.adsabs.harvard.edu/abs/2024esoar.02354794>.
- [57] ARGENE M, RAVENSCROFT C, KINGSWELL I. Ransomware detection via cosine similarity-based machine learning on bytecode representations [EB/OL]. <https://www.authorea.com/doi/full/10.22541/au.172348750.00074165>.
- [58] RAFAPA J, KONOKIX A. Ransomware detection using aggregated random forest technique with recent variants [EB/OL]. <https://www.authorea.com/doi/full/10.22541/au.172426891.14153527>.
- [59] PALAU F, CATANIA C, GUERRA J, et al. DNS tunneling: a deep learning based lexicographical detection approach [DB/OL]. <https://arxiv.org/abs/2006.06122>.
- [60] DING S, ZHANG D, GE J, et al. Encrypt DNS traffic: automated feature learning method for detecting DNS tunnels [C]//*Proceedings of the IEEE International Conference on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing Networking*, 2021: 352–359.
- [61] LIU X, ZHANG Y, YANG X, et al. MFC-DoH: DoH tunnel detection based on the fusion of MAML and F-CNN [C]//*Proceedings of the 21st ACM International Conference on Computing Frontiers*, 2024: 267–275.

(责任编辑: 孙娟)