软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2024,35(5):2522−2542 [doi: 10.13328/j.cnki.jos.006988]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

IPv6 中一种基于卷积的 DDoS 攻击两阶段防御机制*

王郁夫[1], 王兴伟[1], 易 波[1], 黄 敏[2]

[1](东北大学 计算机科学与工程学院, 辽宁 沈阳 110819)

[2](东北大学 信息科学与工程学院, 辽宁 沈阳 110819)

通信作者: 王兴伟, E-mail: wangxw@mail.neu.edu.cn

摘 要: 针对 IPv6 快速普及背景下分布式拒绝服务 (DDoS) 攻击威胁不断增长的现状, 提出一种两阶段的 DDoS 攻击防御机制, 包括初期实时监控 DDoS 攻击发生的预检测阶段, 以及告警后精准过滤 DDoS 攻击流量的深度检测阶段. 首先, 分析 IPv6 报文格式并解析 PCAP 流量捕获文件中的 16 进制头部字段作为样本元素. 其次, 在预检测阶段, 引入轻量化二值卷积神经网络 (BCNN), 设计一种二维流量矩阵作为模型输入, 整体感知网络在混杂 DDoS 流量后出现的恶意态势作为告警 DDoS 发生的证据. 告警后, 深度检测阶段介入, 引入一维卷积神经网络 (1DCNN) 具体区分混杂的 DDoS 报文, 从而下发阻断策略. 在实验中, 自建 IPv6-LAN 拓扑并基于 NAT 4to6 技术重放 CIC-DDoS2019 公开集生成纯 IPv6-DDoS 流量源测试. 结果证明, 所提机制提升针对 DDoS 攻击的响应速度、准确度和攻击流量过滤效率, 当 DDoS 流量出现仅占总网络 6% 和 10% 时, BCNN 就能以 90.9% 和 96.4% 的准确度感知到 DDoS 攻击的发生, 同时 1DCNN 能够以 99.4% 准确率区分 DDoS 报文并过滤.

关键词: DDoS 防御; 两阶段; DDoS 攻击监控; DDoS 流量过滤; BCNN 和 1DCNN; IPv6

中图法分类号: TP309

中文引用格式: 王郁夫, 王兴伟, 易波, 黄敏. IPv6中一种基于卷积的DDoS攻击两阶段防御机制. 软件学报, 2024, 35(5): 2522–2542. http://www.jos.org.cn/1000-9825/6988.htm

英文引用格式: Wang YF, Wang XW, Yi B, Huang M. Two-stage DDoS Attack Defense Mechanism Based on Convolution in IPv6. Ruan Jian Xue Bao/Journal of Software, 2024, 35(5): 2522–2542 (in Chinese). http://www.jos.org.cn/1000-9825/6988.htm

Two-stage DDoS Attack Defense Mechanism Based on Convolution in IPv6

WANG Yu-Fu[1], WANG Xing-Wei[1], YI Bo[1], HUANG Min[2]

[1](School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China)

[2](College of Information Science and Engineering, Northeastern University, Shenyang 110819, China)

Abstract: Aiming at the growing threat of distributed denial of service (DDoS) attacks under the rapid popularization of IPv6, this study proposes a two-stage DDoS defense mechanism, including a pre-detection stage to real-time monitor the early appearance of DDoS attacks and a deep-detection stage to accurately filter DDoS traffic after an alarm. First, the IPv6 traffic format is analyzed and the hexadecimal header fields are extracted from PCAP capture files as detection elements. Then, in the pre-detection stage, a lightweight binary convolutional neural network (BCNN) model is introduced and a two-dimensional traffic matrix is designed as model input, which can sensitively perceive the malicious situation caused by mixed DDoS traffic in the network as evidence of DDoS occurrence. After the alarm, the deep-detection stage will intervene with a one-dimensional convolutional neural network (1DCNN) model, which can specifically distinguish the mixed DDoS packets with one-dimensional packet vector as input to issue blocking policies. In the experiment, an IPv6-LAN topology is built and the proposed pure IPv6-DDoS traffic is generated by replaying the CIC-DDoS2019 public set through NAT 4to6. The results show that the proposed mechanism can effectively improve response speed, detection accuracy, and traffic filtering efficiency in DDoS defense. When DDoS traffic only takes 6% and 10% of the total network, BCNN can perceive the occurrence of

* 基金项目: 国家自然科学基金 (62032013, 62002055)
收稿时间: 2022-12-21; 修改时间: 2023-03-16; 采用时间: 2023-06-09; jos 在线出版时间: 2023-11-08
CNKI 网络首发时间: 2023-11-10

DDoS with 90.9% and 96.4% accuracy, and the 1DCNN model can distinguish mixed DDoS packets with 99.4% accuracy at the same time.

Key words: DDoS defense; two-stage; DDoS attack monitoring; DDoS traffic filtering; BCNN and 1DCNN; IPv6

1 引 言

随着 IT 行业的不断发展, 网络安全面临着更为严峻的挑战, 已不仅是只关乎 IT 行业的问题, 而是逐渐涉及国民经济, 政治, 文化和国防等越来越多的方面. 网络空间国际战略报告[1]指出, 在过去的 8 年中, 全球网络安全事件的数量增加了 11 倍. 而在其中, 分布式拒绝服务 (distributed denial of service attack, DDoS) 攻击在攻击事件中的比例已高达 50.2%. DDoS 攻击是一种针对网络服务的攻击方式, 攻击者通过分布式僵尸网络向目标发送洪水式的服务请求等流量, 耗尽目标主机或服务器的处理能力、带宽资源或服务容量, 最终使目标无法接收或响应来自合法用户的正常服务请求. DDoS 攻击有着攻击范围广, 隐蔽性强, 操作简单, 难以缓解的特点, 成为现今大规模, 有害且难以预防的常见网络攻击之一[2]. 随着 IPv6 网络的逐渐普及, 接入设备的数量也在不断增加, 越来越多的企业和服务提供商开始通过 IPv6 网络提供服务. 如图 1 所示, 截至 2022 年 11 月, Google 使用 IPv6 网络提供服务的用户数量占总用户的比例已达到 39.88%[3], 且按照历史趋势, IPv6 的服务比例在未来势必还会加速增长.

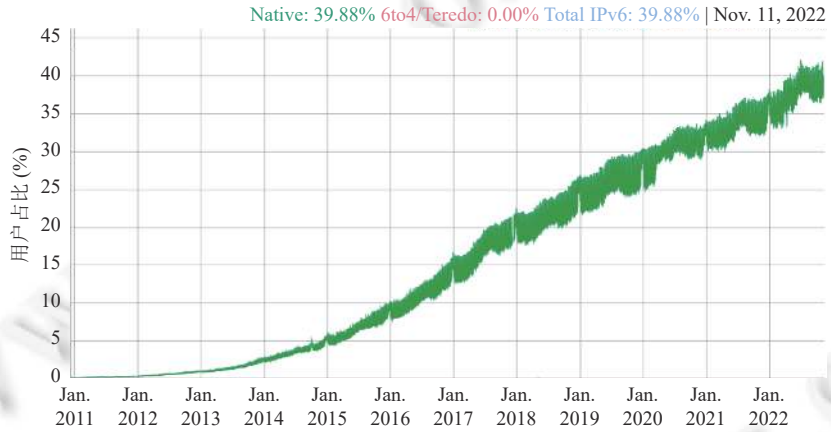


图 1 谷歌通过 IPv6 网络提供服务的用户占比

尽管 IPv6 凭借着 IPSec[4]在安全性上有所保障, 但如图 2 所示, IPv6 网络仍然面临着诸多潜在的安全威胁, 其中 DDoS 攻击威胁的可能性是最高的, 达到了 68%[5], IPv4 网络中的大多 DDoS 攻击手段很容易在 IPv6 网络中发起. 早在 2018 年 3 月, 互联网工程师就发现了业界首个仅基于 IPv6 网络的 DDoS 攻击.

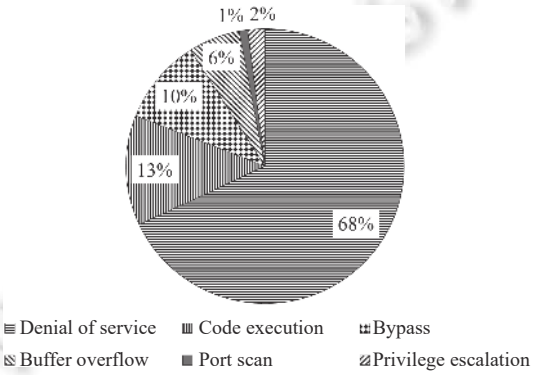


图 2 IPv6 网络潜在安全威胁分布

不仅如此, 随着 IPv6 网络的继续发展, 其超大地址空间势必带来远超现有 IPv4 的巨大网络访问规模, 伴随而来的势必是更猛烈、更复杂的 DDoS 攻击威胁[6]. 截至目前, 有记录的 DDoS 攻击峰值流量已达到 1.45 Tb/s. 往往当受害者意识到正在遭受 DDoS 攻击时, 网络中已然达到的攻击流量规模使得检测器难以招架. 基于报文格式不同, 现有的 IPv4 环境下的防御手段方法在 IPv6 中使用时必然会因为报文信息的不匹配而降低可用性, 同时 IPv6 网络中可预见增长的攻击威胁以及新兴的特殊攻击类型, 比如 ICMPv6[7], 邻居发现协议 (neighbor discovery protocol)[8]等, 迫切需要我们将 IPv6 的特点作为构建 DDoS 威胁模型中的重要部分, 设计针对 IPv6 DDoS 攻击流量更细化的防御机制. 在防御过程中不仅要提升 DDoS 流量的过滤能力, 更应研究如何在 DDoS 攻击发生后更早做出反应, 告警并在攻击规模有限时提早介入, 为组织后续防御抢夺时间以减少威胁. 在本文研究的 DDoS 攻击防御中, 我们主要针对以下两个问题.

1) DDoS 攻击监控问题: 在正常网络中如何监控 DDoS 攻击的发生, 即如何尽早感知 DDoS 导致网络出现的隐蔽恶意特征, 从而对 DDoS 攻击快速告警, 尽早响应介入.

2) DDoS 流量过滤问题: 在监控到 DDoS 攻击发生后, 如何在网络流量中准确区分出混杂的 DDoS 攻击报文, 从而下发阻断策略以过滤攻击.

尽管 DDoS 攻击的手段方法不断更新, 但“流量”仍是 DDoS 攻击的必要手段. 在网络流量信息中, 相较于流级别统计信息不同, 包级别流量信息含有与头部和数据包有效载荷相关的完整信息, 能够详细记录网络活动并支撑实时检测[9]. 因此, 本文以数据包级别信息作为样本元素, 提出一种两阶段的 DDoS 攻击防御机制, 包括预检测阶段中基于二值卷积神经网络 (binary convolutional neural network, BCNN) 的 DDoS 攻击监控机制和深度检测阶段中基于一维卷积神经网络 (one dimensional convolutional neural network, 1DCNN) 的 DDoS 流量过滤机制. 主要贡献归纳如下.

1) 针对 IPv6 流量的预处理部分, 首先, 本文基于 IPv6 报文头部定长等特点, 解析 PCAP 格式捕获的流量文件中存储的 16 进制报文头部字段信息. 其次, 针对预检测阶段的 DDoS 事件监控任务, 本文设计一种 100×82 的二维流量矩阵存储网络中的连续流量信息作为监控分析样本以提升效率. 而对于深度检测阶段的 DDoS 流量过滤任务, 本文设计 1×82 的 1 维报文向量样本, 其中记录单个报文的头部信息.

2) 本文设计了一种两阶段的 DDoS 防御策略, 将防御细化为监控 DDoS 攻击发生的预检测阶段和过滤 DDoS 攻击流量的深度检测阶段. 这样, 在网络中发生 DDoS 攻击时, 预检测阶段能在网络中仅出现小规模的 DDoS 攻击流量时感知恶意的流量态势变化, 从而快速告警响应启动深度检测阶段具体区分网络中混杂的 DDoS 报文, 下发阻断策略过滤攻击.

a) 在预检测阶段中, 本文引入一种轻量化的 BCNN 深度学习模型作为决策核心, 以二维流量矩阵作为模型输入对网络流量进行大尺度的粗粒度监控. 当流量矩阵采样到包含零星 DDoS 报文的网络流量片段时, BCNN 能够灵敏感知流量态势的恶意变化, 从而告警此时网络中出现 DDoS 攻击.

b) 在深度检测阶段中本文同样考虑检测的轻量化, 引入 1DCNN 模型作为决策核心, 将报文头部字段组成的一维向量作为输入样本. 1DCNN 模型特殊的一维卷积结构能够准确识别当前报文是否为 DDoS 攻击报文, 从而下发阻断策略进行过滤.

3) 在实验中, 为了构建真实的 IPv6 网络实验环境, 本文在 CERNET2[10]东北大学校园网的环境下搭建 IPv6-LAN 拓扑, 并在其上验证本文提出机制. 首先, 为了解决现如今研究中 IPv6 环境公开 DDoS 流量集匮乏的问题, 本文引入一种 IPv4 环境新颖的公开数据集 CIC-DDoS2019[11], 利用 Tcpreplay 重放技术[12]结合 NAT 4to6 翻译技术[13]在 IPv6-LAN 拓扑中实现 IPv6-DDoS 流量重放. 其次, 在此基础上, 我们测试了预检测阶段中基于 BCNN 模型的 DDoS 攻击监控机制和基于 1DCNN 模型的流量过滤机制在准确度、查准率、*F*1-*score*、ROC 等指标上的表现, 并与现有的解决方案进行对比以体现优势. 最后, 本文在 IPv6-LAN 拓扑中模拟完整的 DDoS 攻击防御任务, 通过对比 DDoS 攻击的缓解曲线验证所提出的两阶段策略在 DDoS 流量过滤效率上的优势.

本文第 2 节回顾相关工作. 第 3 节介绍系统框架. 在第 4 节和第 5 节分别介绍流量预处理与两阶段防御策略. 第 6 节对提出机制进行实验测试. 第 7 节总结本文工作.

2 相关工作

在本节中, 介绍了 DDoS 攻击事件监控和流量过滤领域中的现有检测方法, 包括传统解决方法和基于人工智能的解决手段, 并分析了现阶段工作亟待提升的方面.

2.1 DDoS 攻击事件监控研究

在 DDoS 攻击防御工作中, 很少有针对 DDoS 攻击事件监控方面的研究. 但随着 DDoS 攻击威胁日益增加, 在 DDoS 发生后快速响应, 及时向网络管理员发出警报, 尽早采取措施往往能够大幅减轻 DDoS 攻击的危害并为防御工作争取时间.

早期工作中, Yuan 等人[14]提出了一种检测早期 DDoS 攻击事件的方法, 通过在网络中设置的互相关观察点, 监控 DDoS 攻击在宏观上引起的时空流量模式的变化. Singh 等人[15]监控流经边缘路由器的流量, 通过设置归一化路由器熵、数据包速率和熵速率阈值实时监控 DDoS 行为. Liu 等人[16]以模糊控制理论为基础, 结合统计理论建立正常传输行为模型, 以实现网络监控和预警功能. Liu 等人[17]提出名为 UnivMon 流量监测框架, 基于草图理论使用数据平面中的流统计信息来计算应用程序级指标, 并验证了其在一系列监控任务中实现了相对于自定义草图方案更优秀的准确性. Wang 等人[18]基于软件定义网络 (software defined network, SDN) 的流量全局信息提出一种 DDoS 攻击缓解架构, 引入高度可编程的网络监控以及灵活的控制结构实现快速和准确的攻击响应. Galeano-Brajones 等人[19]提出一种基于熵的解决方案, 使用有状态的数据平面监控物联网场景中的 DDoS 攻击. Xie 等人[20]提出了一种基于隐半马尔可夫的新型异常检测器来描述访问矩阵的动态变化并检测攻击. Biswas 等人[21]基于虚拟机 (virtual machine, VM) 进行分层聚类, 提出了一种基于 VM 间行为相似性的流分组方法, 提升实时监控性能. Baskar 等人[22]提出了一种使用多阈值流量分析的新型实时流量监控算法, 分析网络中有效载荷、跳数、延迟、数据包数等信息实现实时 DDoS 监控. Zaib 等人[23]在实时监控问题中引入神经网络思想, 使用人工神经网络和支持向量机 (support vector machine, SVM) 进行分类.

从上述工作看出, DDoS 攻击的监控手段正在不断地更新, 从原始的宏观观察逐渐发展到基于 IP 熵等流量统计学指标. 再到针对流量数据引入的聚类、马尔可夫链、SVM 等人工智能分析手段. 下一步, 我们应该引入更为优秀, 更贴合 DDoS 攻击监控场景的人工智能核心, 探讨监控实时性的提升.

2.2 DDoS 攻击流量过滤研究

DDoS 攻击流量过滤工作的重点在于对网络中混杂的 DDoS 流量报文进行区分, 从而在监控到 DDoS 攻击发生后, 有效提取网络中 DDoS 流量的来源并基于此下发针对性的流量阻断策略以缓解攻击. 近几年, 大量引入人工智能模型作为检测核心的 DDoS 流量识别工作成为主流.

Yuan 等人[24]提出了一种基于深度学习的 DDoS 攻击检测方法 DeepDefense, 主要基于循环神经网络 (recurrent neural network, RNN) 实现高级特征提取, 相较于传统机器学习方法有着更高准确度. Rehman 等人[25]提出一种名为 DIDDOS 的新型高效策略, 使用基于门控循环单元 (gated recurrent unit, GRU) 的 RNN 网络抵御现实世界中的新型 DDoS 攻击. Ali 等人[26]提出了一种快速、大规模的监控系统, 获取在 DDoS 攻击发生时的暗网流量并形成数据集进行训练并用于检测. Saad 等人[27]在 IPv6 网络中提出使用反向传播神经网络的智能 ICMPv6 DDoS 攻击检测框架 v6IIDS. Ye 等人[28]构建基于 Mininet 和 floodlight 仿真平台的 SDN 环境, 提取交换机流量表的 6 元特征值, 结合 SVM 分类算法建立 DDoS 攻击模型. Parra 等人[29]提出基于云的分布式深度学习框架, 包括两个协同工作的关键安全机制: (1) 作为物联网设备微安全插件的分布式卷积神经网络模型; (2) 基于云计算的时间长短期记忆网络模型, 以实现钓鱼和僵尸网络攻击检测和缓解. Premkumar 等人[30]提出一种新的轻量级 DoS 检测方案 DLDM, 基于深度学习来检测和隔离数据转发阶段的攻击. Doriguzzi-Corin 等人[31]提出一种实用的、轻量级的深度学习 DDoS 检测系统 Lucid, 利用卷积神经网络检测流量. Asad 等人[32]提出使用前馈反向传播可以准确检测多种应用层 DDoS 攻击. Hwang 等人[33]提出一种有效的异常流量检测机制, 即 D-PACK, 该机制由 CNN 和无监督深度学习模型 (如 Autoencoder) 组成, 自动分析流量模式并过滤异常流量. Cil 等人[34]基于深度神经网络 (deep neural

network, DNN) 设计检测模型, 通过分析网络流量中捕获的样本数据包进行检测. DNN 模型在结构上包含了特征提取和分类过程, 且在训练过程中能够自我更新, 因此即使在小样本下也能快速、高精度地工作. Wei 等人[35]结合 CNN 和 RNN, 提出一种基于深度学习的分层时空特征学习异常检测方法 HAST-NAD, 能够自动学习流量特征, 提高流量异常检测效率.

相比于传统方法, 深度模型可以检测的攻击类型广泛, 准确率高, 时间序列感知力强, 前期工作简单[36]. 但是, 随着 DDoS 流量识别的网络场景的复杂化, 深度学习模型的使用应考虑轻量化问题, 尝试引入结构更轻、开销更低的新颖技术来解决问题.

3 总体设计

本文的两阶段防御机制结构如图 3 所示包含 4 个部分: IPv6-SAVI 实验拓扑, IPv6 流量预处理策略, 检测模型构建和两阶段 DDoS 攻击防御策略.

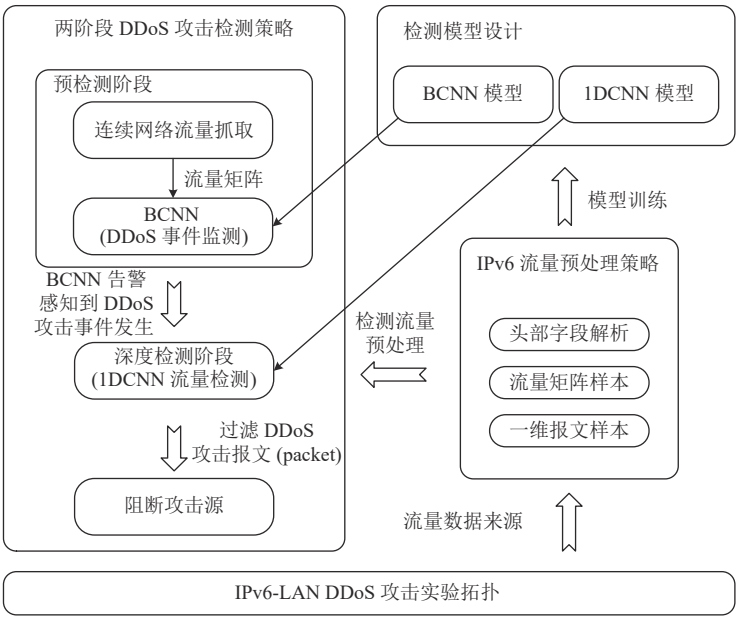


图 3 系统结构图

1) 本文在 CERNET2 东北大学校园网中的服务器上搭建 IPv6-LAN 拓扑作为实验环境. 在其上, 我们引入 IPv4 公开流量集 CIC-DDoS2019, 借助 NAT 4to6 翻译技术在 IPv6-LAN 拓扑中重放来模拟纯 IPv6 的 DDoS 攻击流量源, 以此解决 IPv6 环境公开 DDoS 流量集的匮乏问题.

2) IPv6 流量预处理策略旨在将网络中捕获的原始流量分析处理成可输入深度学习模型的样本格式和内容. 在预处理中, 本文对 IPv6 报文头部字段解析并分别设计流量矩阵和一维报文两种输入样本格式, 分别对应 BCNN 模型和 1DCNN 模型的输入.

3) 检测模型设计旨在基于 DDoS 防御中的事件监控和流量过滤的不同任务需求构建两种不同特点的卷积神经网络: BCNN 和 1DCNN 作为决策核心.

4) 两阶段的 DDoS 攻击防御策略能够在 IPv6 流量预处理策略和检测模型设计的基础上实现 DDoS 攻击的快速防御任务, 包括预检测阶段和深度检测阶段. 其中, 预检测阶段利用 BCNN 模型的矩阵视野高效监控网络流量, 当 DDoS 流量出现时感知恶意的态势变化来告警当前网络中发生 DDoS 攻击. 而深度检测阶段则在告警后, 利用 1DCNN 模型逐条区分和过滤网络流量中混杂的 DDoS 报文.

接下来, 我们对以上部分进行细节介绍.

4 IPv6 流量预处理策略

4.1 IPv6 报文头部信息提取

IPv6 流量预处理策略旨在将网络中抓取到的原始 IPv6 流量数据处理成可输入本文构建的深度学习检测模型的格式. 在实验中, 我们使用 Wireshark 作为流量捕捉器, 以 PCAP 格式抓取网络中的 IPv6 流量, 然后解析存储报文的头部信息进行分析. 为了提升流量信息的提取速度, 本文基于 IPv6 报文头部的定长特点设计解析流程, 直接提取 PCAP 中的 16 进制原始报文字段作为样本数据, 省去传统的报文信息转译等步骤[37].

图 4 中, 本文举例展示了抓取的 PCAP 流量文件中从头开始存储的 16 进制数据. 其中, PCAP 文件是以一个 PCAP 文件头开始, 它记录了当前 PCAP 文件捕获流量的时间跨度、报文数量等信息, 如图 4 中单下划线部分, 固定长度为 24 B, 之后是依次存储的报文 (packet) 数据. 图 4 中, 虚线包围部分为当前 PCAP 记录的第 1 个报文. 每个报文的信息由两部分组成: 双下划线标记的帧信息部分, 长度固定为 16 B, 用于记录包的到达时间、经历时间和捕获长度等信息; 以及波浪下划线标记的内容部分, 包括报文的头部信息和负载数据, 长度是可变的.

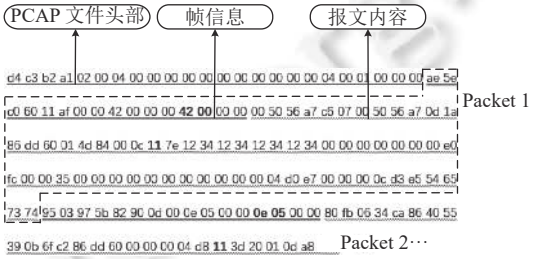


图 4 PCAP 存储报文内容

基于此, 我们切分 PCAP 存储的报文并提取头部字段的流程如下. 跳过 24 B 的 PCAP 文件头部, 在第 1 个报文中, 帧信息部分的第 13 和 14 字节以倒序存储当前报文内容的长度, 如图 4 中, “0042”, 即 66 B. 因此, 我们从帧尾部向后取 66 B 即可得到当前报文的所有内容, 同时获得下一个报文的帧的起始位置. 基于报文信息第 21 字节匹配当前报文的类型, 比如 packet 1 中加粗的“11”表示当前报文类型为 ICMPv6. 这样, 结合 IPv6 不同报文类型不同的头部定长特点, 即可在报文内容中截取对应报文的完整头部信息. 本文中, 我们研究了 ICMPv6、TCP 和 UDP 这 3 种报文类型, 也是 DDoS 防御中最常见分析的 3 种. 表 1 展示了上述 3 种类型报文的不同字段及在头部中的对应位置, 包括 eth 和 IPv6 的共有字段及各类型报文中的特殊字段.

表 1 IPv6 报文头部字段信息

字段	ICMPv6头部		TCP头部		UDP头部	
	字段名称	位置	字段名称	位置	字段名称	位置
共有字段	eth.dst	B (1–6)	ipv6.tclass.dscp	B 15 (2/2)	ipv6.nxt	B (21)
	eth.addr	B (7–12)	ipv6.tclass.ecn	B 16 (1/2)	ipv6.hlim	B (22)
	eth.type	B (13–14)	ipv6.flow	B ((2/2)–18)	ipv6.src	B (23–38)
	ipv6.version	B 15 (1/2)	ipv6.plen	B (19–20)	ipv6.dst	B (39–54)
特殊字段	icmpv6.type	B (55)	tcp.srcport	B (55–56)	udp.srcport	B (55–56)
	icmpv6.code	B (56)	tcp.dstport	B (57–58)	udp.dstport	B (57–58)
	icmpv6.checksum	B (57–58)	tcp.seq	B (59–62)	udp.length	B (59–60)
	icmpv6.identifier	B (59–60)	tcp.ack	B (63–66)	udp.checksum	B (61–62)
	icmpv6.sequence_number	B (60–62)	tcp.hdr_len	B (67(1/2))	udp.payload	B (63–end)
	icmpv6.payload	B (63–end)	tcp.flags	B (67(2/2)–68)		
			tcp.window_size_value	B (69–70)		
			tcp.checksum	B (71–72)		
			tcp.urgent_pointer	B (73–74)		
			tcp.payload	B (75–end)		

4.2 模型输入样本设计

图 5 所示, 在本文中, 我们将 IPv6 正常流量和 DDoS 攻击流量混合作为数据源. 针对 BCNN 模型的二维矩阵输入和 1DCNN 模型的一维向量输入, 设置了名为流量矩阵和一维报文两种不同的样本.

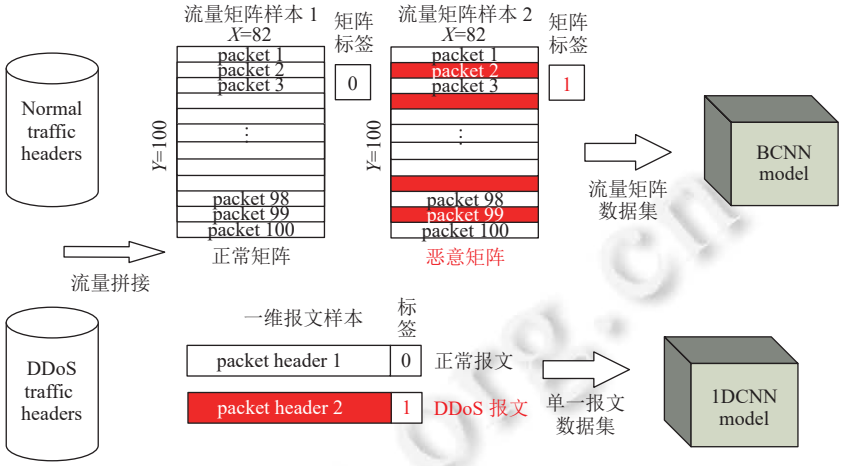


图 5 流量矩阵样本和一维报文样本

1) 流量矩阵样本: 在预检测阶段, 本文通过监控网络中是否出现 DDoS 攻击流量来监控 DDoS 攻击. 显然, 在正常网络流速背景下, 逐一检测网络报文进行监控是不可行的, 这不仅效率极低且会带来难以想象的巨大开销. 因此, 本文提出一种如图 6 所示的 100 行 82 列的二维流量矩阵来收集流量. BCNN 模型将流量矩阵整体作为输入, 一次性扫描矩阵中的 100 行报文. 当 DDoS 攻击流量出现时, BCNN 模型将感知矩阵异常的流量态势变化作为目标, 间接判断网络中有无 DDoS 攻击流量的存在. 这样, 本机制能在网络中高效监控流量, 并在 DDoS 攻击出现时快速告警.

Number	1–6	7–12	13–14	15	16–18	19–20	21	22	23–38	39–54	55	56	57–58	59–60	61–64	65–68	69	70	71–72	73–74	75–76	77–78	79–80	81–82
X_name / Y_name	eth. dst	eth. addr	eth. type	ver & dsc	ipv6. flow	ipv6. plen	ipv6. nxt	ipv6. hlim	ipv6. src	ipv6. dst	icmpv6. type	icmpv6. code	tcp/udp. src. port	tcp/udp. des. port	tcp. seq	tcp. ack	tcp. hdr_len	tcp. flags	tcp. Win dow_size	udp. length	icmpv 6/ud pcheck	icmpv6. identifier	icmpv6. sequence	tcp. urgent_pointer
packet 1																								
packet 2																								
packet 3																								
packet 4																								
...																								
packet 100																								

图 6 流量矩阵样本

流量矩阵具有 100×82 的结构, 其中 Y 表示每个矩阵中包含从网络中连续抓取的 100 个流量报文的头部信息, X 表示每个报文中的不同头部字段. 本文在流量矩阵中共设置了 82 个头部字段, 其中 X_1–X_{54} 是公共头部字段, 其他部分是特殊字段, 用于满足在同一个流量矩阵中集合 ICMPv6、TCP 和 UDP 这 3 种不同类型的报文头部信息. 比如, ICMPv6 占用 $X_{55,56,75–80}$, TCP 占用 $X_{57–72,81–82}$, UDP 占用 $X_{57–60,73–76}$. 在形成流量矩阵时, 我们根据捕获的每个报文的协议类型将头部字段填充到每一行的相应位置, 最后用 0 填充空的字段.

对应的, 如图 5 所示, 本文基于流量矩阵样本构建数据集, 由<流量矩阵样本, 矩阵标签>构成. 其中, 我们将包含 DDoS 攻击报文的流量矩阵打标签为 1: 恶意, 而将全部为正常报文的流量矩阵打标签为 0: 正常. 同时, 本文通过调整流量矩阵中 DDoS 攻击报文的个数来模拟发生不同强度 DDoS 攻击事件[38]时的网络流量片段, 从而在实验中测试 BCNN 模型的监控精度变化.

2) 一维报文样本: 在深度检测阶段, 本文使用 1DCNN 模型来具体区分网络中的单个报文是否为 DDoS 攻击

流量. 因此, 1DCNN 的数据集由<一维报文样本, 报文标签>构成. 我们参照流量矩阵中 X 轴上的 82 个字段, 针对每个报文形成 1×82 的固定一维向量作为 1DCNN 模型的输入. 其中, 标签 (0: 正常, 1: 恶意) 表示当前报文是否为 DDoS 攻击流量.

5 两阶段的 DDoS 防御策略

5.1 两阶段防御流程

近年来, 两阶段策略被广泛应用在目标检测中[39], 其中, 第 1 阶段会完成对多个候选区域的粗粒度挑选, 第 2 阶段则会在候选区域内进行细粒度检测. 本文借鉴这种两阶段的目标检测思想, 提出一种两阶段的防御策略, 将 DDoS 攻击防御细化为预检测阶段和深度检测阶段, 旨在提升 DDoS 攻击的响应速度以及流量过滤效率, 如图 7 所示.

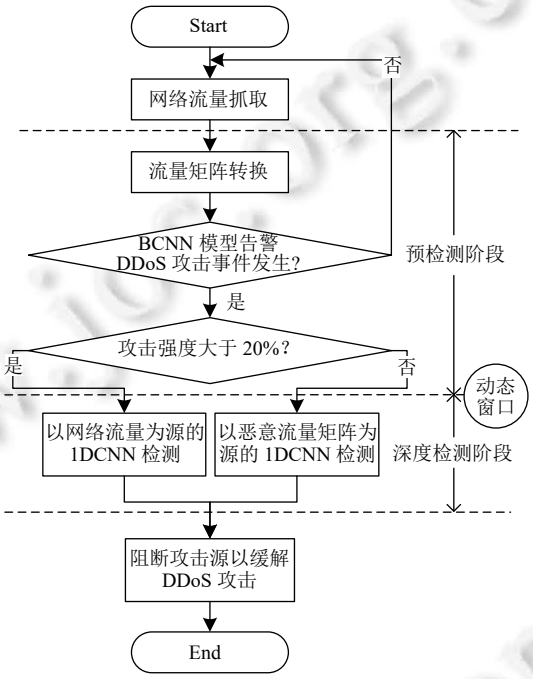


图 7 两阶段 DDoS 防御策略

预检测阶段完成的是网络中 DDoS 攻击事件的监控任务. 即在正常网络中, 预检测阶段能够利用 BCNN 模型的矩阵视野高效监控网络流量. 当 DDoS 攻击事件发生后, BCNN 模型能够在网络中出现少量 DDoS 流量时, 感知到其导致的网络流量态势的恶意变化, 从而灵敏告警发生尚处早期的 DDoS 攻击事件. 深度检测阶段则在告警后尽快介入, 使用 1DCNN 模型检测一维报文样本, 区分出网络中混杂的 DDoS 攻击报文, 提取源信息进行阻断缓解. 特别的是, 在预检测阶段告警后, 本文将计算当前网络中出现的 DDoS 攻击事件的“强度”, 即由 1DCNN 模型对当前的恶意流量矩阵中的全部报文进行检测并统计其中 DDoS 攻击报文个数所占百分比. 在理想条件下, DDoS 强度应等同于 DDoS 流量速率占网络中总流量速率的百分比. 本文在实验中发现, 当 DDoS 强度小于 20% 时, 从网络中截取的流量矩阵中有概率不包含攻击报文, 即有概率抓取到连续的 100 个正常报文. 因此, 我们以 20% 作为阈值, 当 DDoS 强度大于 20% 时, 深度检测阶段会直接从网络中抓取流量进行检测. 当攻击强度小于 20% 时, 深度检测阶段会以预检测阶段识别为恶意的流量矩阵中记录的报文作为源. 基于此, 深度检测阶段在缓解降至较低强度的 DDoS 攻击时, 每检测 100 条网络报文, 均能够有效地命中 DDoS 报文, 减少无用开销以提升 DDoS 攻击缓解效率. 同时, 显然在每个深度检测阶段前都对 DDoS 攻击事件进行强度判定会带来冗余的判断, 因此本文设计一

种动态窗口策略. 首先, 预检测阶段在告警后会以时间窗口 T 作为间隔监控网络流量, 而 T 的大小与当前强度与 20% 的差值成正比, 即在强度较高或较低阶段, 远大于 20% 或小于 20% 时较大, 而在接近 20% 时减小, 从而在实现灵敏的深度检测逻辑切换的同时减少检测冗余. 最后设置计数器 C, 即在 C 个时间窗口 T 内预检测阶段连续监控到网络中无 DDoS 攻击时, 结束检测.

5.2 预检测阶段-BCNN 决策核心

在预检测中, 我们以流量矩阵作为样本, 对网络中 DDoS 流量的出现进行动态监控. 为了实现这种针对矩阵的分析, 初期引入卷积神经网络作为决策核心的基础结构 (Base-CNN). 在此基础上, 针对监控问题在网络中日常化的长时间运行特点, 本文进一步引入二值化轻量卷积核, 结合全局平均池化 (global average pooling, GAP) 等模型压缩方法, 构建更加高效率、低开销的 BCNN 深度学习决策核心, 如表 2 所示.

表 2 Base-CNN 和 BCNN 模型结构

Base-CNN			BCNN		
Layername	Output size	8 layers	Layername	Output size	12 layers
Input	100×82	—	Input	100×82	—
Conv_1_x	94×76, 2	7×7, 2, stride 1	Conv_1	94×76, 4	7×7, 4, stride 1
Conv_2_x	94×76, 4	5×5, 4, stride 1 5×5, 4, stride 1	Binary_conv1_x	47×38, 8	5×5, 8, stride 1 5×5, 8, stride 1 2×2 max pooling, stride 2
Conv_3_x	94×76, 8	3×3, 8, stride 1 3×3, 8, stride 1	Binary_conv2_x	23×19, 16	3×3, 16, stride 1 3×3, 16, stride 1 2×2 max pooling, stride 2
Flatten	57 152	—	Binary_conv3_x	11×9, 16	3×3, 16, stride 1 3×3, 16, stride 1 3×3, 16, stride 1
Dense	2	—	GAP	16	global average pool, stride 2
			Dense	2	—

在最初的 Base-CNN 设计上, 我们从两层普通卷积神经网络开始, 逐步增加卷积深度、卷积核个数及通道数, 提高模型的识别能力. 通过性能的不断逼近, Base-CNN 包含 8 层, 包括输入层、一个 7×7、两个 5×5、两个 3×3 卷积层、一个 Flatten 层和一个二分类的 Dense 层, 共包含 113 566 个参数, 在实验中面对流量矩阵中 DDoS 攻击流量占比 10% 时能够达到 98.3% 的识别准确率.

基于此, 表 2 中我们确定 BCNN 结构并在图 8 中画出了 BCNN 的模型结构图. 本文的 BCNN 模型由 13 层构成, 包括输入层、1 个普通的 7×7 卷积层、2 个 5×5 的二值卷积层、4 个 3×3 的二值卷积层、3 个池化层、1 个全局平均池化层和 1 个用于二分类的 Dense 层. 相较于 Base-CNN 结构, BCNN 模型的轻量化主要体现 3 方面.

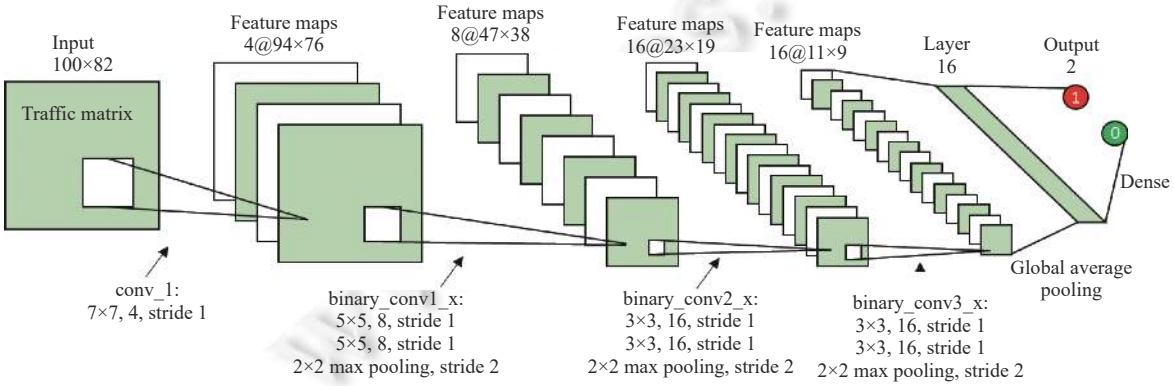


图 8 BCNN 模型结构图

1) 全连接参数压缩: Base-CNN 作为普通卷积模型结构, 其包含 98.5% 的全连接参数, 这是第 1 个压缩的重点. 我们使用 GAP 替换原本的 Flatten 层, 将输入到 Dense 层的变量数量从 57 152 减少到 16. 这样, 模型的参数量从 113 566 下降到 11 114.

2) 卷积层参数压缩: 在全连接部分参数量压缩后, 我们进一步针对卷积层部分进行轻量化操作. 本文在 DDoS 检测任务解决方案中创新性地引入了二值化的思想[40]. 在训练的正向传播中通过确定性二值化公式将浮点型卷积核参数 x 二值化为 1 bit 存储, 即将所有 x 大于等于 0 存储为+1, 其余 x 存储为−1, 并在训练中结合算法 1 的参数更新策略形成二值化卷积层 (Binary_conv).

算法 1. 二值化卷积核参数前向传播.

输入: 上一层的激活值 $a^b_{(k-1)}$, 本层权值 w^k;
输出: 本层的激活值 a^b_k.

1. BEGIN
2. FOR k=1 to L /*从第 1 层到第 L 层进行循环*/
3. 对权值 w^k 进行二值化处理, 得到二值化后的权值 w^b_k
4. w^b_k 与上一层二值化后的激活值 $a^b_{(k-1)}$ 相乘得到 S_k
5. 将 S_k, θ_k 进行 BatchNormalization 得到本层的激活值 a_k
6. IF 还未进行到最后一层 THEN
7. 对 a_k 进行二值化得到本层二值化后的激活值 a^b_k
8. END IF
9. END FOR
10. END

在 DDoS 攻击的日常化监控场景中, 二值化后的卷积核参数可以带来以下两个重要优势.

a) 模型大小和运行内存开销的显著减少: Base-CNN 的卷积核参数以 32 位浮点型存储, 但二值化后以 1 bit 二进制存储. 因此, 可以使模型整体存储大小和运行内存开销缩减为原本的 1/32.

b) 计算复杂度的显著降低: 在 Base-CNN 模型中, 卷积核参数与输入矩阵中变量的计算是两个浮点型之间的计算, 而 BCNN 模型的计算则是二进制卷积核参数与浮点型变量之间, 时间复杂度可降低 60%[41], 提升模型运行效率.

3) 经过以上两步轻量化后, 模型的准确性严重下降. 原因在于无论 GAP 还是二值化卷积核参数都会损失模型的分析能力. 因此, 本文在压缩同时增加了二值化卷积层的深度, 将通道数量增加一倍, 并末尾添加了两个 3×3 的卷积层, 使卷积层深度增加到 7 层. 最后, 在每两个二值化卷积层后增加最大池化层, 突出强特征的重要性, 避免模型的过拟合问题.

5.3 深度检测阶段-1DCNN 模型设计

深度检测阶段旨在从网络流量中精准区分混杂的 DDoS 攻击报文. 不同于预检测阶段对网络流量态势的粗粒度分析, 深度检测阶段针对的是网络中的单一数据包, 旨在细粒度区分其是否为 DDoS 攻击产生的恶意流量. 在检测中, 本文以前文图 5 中设计的一维报文样本作为输入, 设计一种基于 1DCNN 的深度学习检测模型. 相比于传统的 RNN、LSTM、DNN 等神经网络模型, 1DCNN 结构有着参数量少, 特征提取能力强等特点, 不仅检测性能优秀, 同时能够减小计算复杂度和开销. 本文在表 3 中给出设计的 1DCNN 模型结构, 并在图 9 中给出模型结构图.

如图 9 所示, 1DCNN-DDoS 共包含 10 层, 包含输入层、卷积核大小为 1×7、1×5、1×3 的 1 维卷积层 4 个, 3 个池化层, 实现轻量降维的 1 个 GAP 层以及用于分类的全连接 Dense 层. 不同于常见的二维卷积核, 1DCNN-DDoS 中卷积核均为 1 维格式, 即 1×n 大小. 在计算中, 我们向 1DCNN 模型输入单个报文头部字段组成的 1×82 的一维向量, 卷积核会在输入向量上按照设定步长横向移动进行特征提取, 最终利用 Softmax 输出二分类结果, 即当前报文是否为 DDoS 攻击报文.

表 3 1DCNN 模型结构

Layername	Output size	10 layers
Input	1×82	—
conv1d_1	1×76, 2	1×7, 2, stride 1
conv1d_2	1×38, 4	1×5, 4, stride 1 1×2 max pooling, stride 2
conv1d_3	1×19, 8	1×3, 8, stride 1 1×2 max pooling, stride 2
conv1d_4	1×9, 8	1×3, 8, stride 1 1×2 max pooling, stride 2
Global average pooling	8	—
Dense (Softmax)	2	—

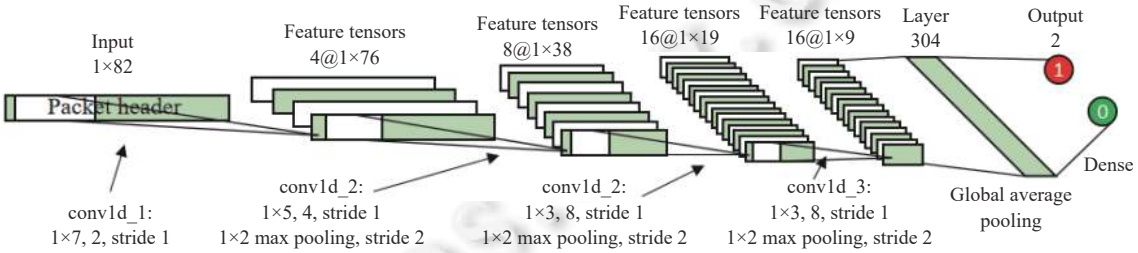


图 9 1DCNN 模型结构图

6 实验验证

在本节中, 本文首先介绍了搭建的 IPv6-LAN 实验拓扑以及实验所需流量源. 进而基于第 4 节的流量预处理策略和样本形式构建实验数据集. 最后, 本文基于二分类经典评价指标测试提出的 BCNN 和 1DCNN 模型在 DDoS 攻击事件监控以及流量过滤任务上的性能, 并模拟实现 DDoS 攻击的完整防御流程来评价本文提出的两阶段策略的表现.

6.1 IPv6-LAN: 实验拓扑

本文在 IPv6 环境下的 CERNET2 东北大学校园网上运行的服务器中搭建如图 10 所示的网络拓扑 (IPv6-LAN) 作为本文的实验环境, 包括 5 种类型的 7 台虚拟主机作为网络拓扑中的关键节点.

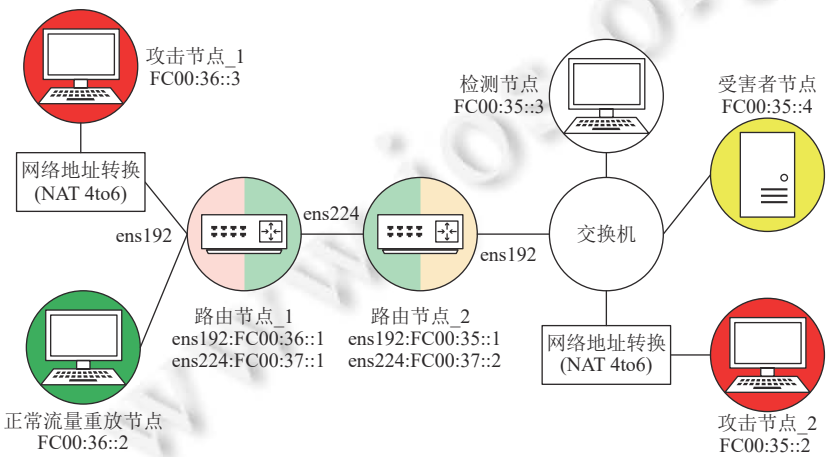


图 10 IPv6-LAN DDoS 攻击模拟网络拓扑

1) 攻击节点_1&2: 用于向受害者发送 DDoS 攻击流量的节点. 我们设置了与受害者在相同和不同的网段中的两个不同攻击节点以扩展流量的多样性. 在攻击节点上, 本文使用 Tcpreplay 技术将 CIC-DDoS2019 公开数据集中保存的 DDoS 攻击流量重放到 IPv6-LAN 中. 但 CIC-DDoS2019 是基于 IPv4 的, 因此, 在流量重放进入 IPv6-LAN 拓扑之前, 我们引入 NAT 4to6 将数据包进行翻译转换, 其静态地址转换如表 4 所示. 经过分析, CIC-DDoS2019 中的 DDoS 流量均为从 172.16.0.5 发送至 192.168.50.1. 但固定的源地址信息显然不符合 DDoS 攻击流量的分布式特点, 因此, 在 Tcpreplay 重放时, 我们首先随机重写 IPv4 的源地址, 进而在 NAT 翻译时将 IPv4 地址以 16 进制拼接在攻击节点所在的网段 FC00:35/36 之后, 形成纯 IPv6 DDoS 攻击流量. 这样, 我们通过调整流量重放的速率能够改变 DDoS 攻击事件的强度.

2) 路由节点_1&2: 即拓扑中的可编程软路由节点, 用于实现不同网段间 IPv6 流量的路由转发.

3) 正常流量重放节点: 在 DDoS 攻击模拟过程中, 该节点可为 IPv6-LAN 输入稳定的正常流量作为背景流量. 我们通过东北大学网络中心在 CERNET2 校园网出口上抓取存储一定量的 IPv6 业务流量并通过 Tcpreplay 重放至 IPv6-LAN 中. 因为校园网内部有着严格的网络监控和安全审查制度, 因此, 这些获取的背景流量可以被视作不含攻击的正常业务流量. 为了满足 IPv6 流量在拓扑中的正常转发, 在流量重放过程中, 我们首先重定向目的节点为“FC00:35::4”, 其次我们替换所有数据包的 IPv6 源地址前两段为“FC00:36”以匹配网段通讯要求.

4) 受害者节点: 作为 DDoS 模拟攻击的目标, 用于接收 DDoS 攻击流量和背景流量.

5) 检测节点: 用于捕获和实时监控目标节点的流量情况. 本文通过在路由节点_2 上配置基于 ip6tables 的流量镜像命令, 将发送到目标主机的流量镜像至检测节点, 进而在检测节点上收集量并部署 DDoS 攻击事件监控和流量检测功能.

具体实验设备参数如表 5 所示.

表 4 NAT 4to6 静态地址转换

CIC-DDoS2019	Tcpreplay	NAT 4to6
源地址	***.***.***.*** 举例: 172.16.0.5	FC00:36::+16进制IPv4地址段 举例: FC00:36::AC::10:0:5 (攻击节点_1) FC00:35::AC::10:0:5 (攻击节点_2)
目的地址	192.168.50.1	FC00:35::4

表 5 实验环境

环境	参数
服务器	Dell PowerEdge R470
虚拟节点环境	OS: Ubuntu 18.04.1 CPU: Intel(R) Xeon(R) Gold 6238R 2.20 GHz RAM: 8 GB
检测节点环境	OS: Ubuntu 18.04.1 CPU: Intel(R) Xeon(R) Gold 6238R 2.20 GHz RAM: 16 GB GPU: NVIDIA GeForce GTX 1080 16 GB
项目环境	Python 3.7.6, TensorFlow-GPU 1.14.0, Keras 2.2.5, Cuda 10.0.130, Cudnn 7.6.5
DDoS攻击仿真平台	Hyenae-ng 1.2
网络流量捕获软件	Wireshark 3.4.5

6.2 实验流量源和训练样本划分

在 IPv6-LAN 上, 本文基于正常流量和 CIC-DDoS2019 的重放捕获实验所需的 IPv6 正常流量和 DDoS 流量源, 结合第 4 节的 IPv6 流量预处理策略分别为 BCNN 模型和 1DCNN 模型构建流量矩阵数据集和单一报文数据集.

在实验中针对模型进行训练和测试时, 数据集中样本的分布和划分如图 11 所示. 无论是流量矩阵数据集或是单一报文数据集, 每个数据集中均包含 10 000 个样本和标签, 包含 50% 的正常样本和 50% 的 DDoS 攻击恶意样本. 本文按照 7:2:1 的比例将数据集划分为训练集、验证集和测试集, 其中训练集和验证集会在模型训练过程中用于学习和参照, 而测试集将仅用于模型训练完毕后的性能测试. 通过对比模型在验证集和测试集上的性能表现, 可以观察模型的泛化能力和过拟合情况.

6.3 BCNN 和 IDCNN 性能评价指标

本文测试了提出的基于 BCNN 的 DDoS 事件监控机制和基于 1DCNN 的 DDoS 流量过滤机制的性能. 评价指

标包括模型训练的损失值 Loss: 模型输出与真实结果之间的距离. 以及基于模型对样本的预测值与样本真实值比较出现的 4 种判断结果: *TP* (被预测为正类的正样本)、*TN* (被预测为负类的负样本)、*FP* (被预测为正类的负样本)、*FN* (被预测为负类的正样本) 得到准确度、*F*1-*score*、查准率、召回率、ROC 曲线以及 AUC:

图 11 数据集样本数量划分

准确度 (*Acc*): 如公式 (1), 表示所有样本中正确预测的比例.

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

查准率 (*Pre*): 如公式 (2), 表示预测正样本中实际为正样本的比例.

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

召回率 (*Recall*): 如公式 (3), 表示实际正样本中预测为正样本的比例.

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

*F*1-*score*: 一种经典的分类问题的性能衡量指标, 如公式 (4) 是查准率和召回率的调和平均数, *F*1-*score* 越大, 模型的性能越好.

$$F1\text{-}score=\frac{2Pre\times Recall}{Pre+Recall} \tag{4}$$

ROC 曲线: 接收者操作特征 (receiver operating characteristic) 曲线, 如公式 (5) 所示, 它以 *FPR* 为横坐标, *TPR* 为纵坐标, 以概率为阈值评估模型性能. ROC 曲线下的面积 (area under curve, AUC) 越大, 说明该模型的性能越好.

$$FPR=\frac{FP}{FP+TN},\ TPR(Recall)=\frac{TP}{TP+FN} \tag{5}$$

6.4 基于 BCNN 的 DDoS 攻击监控机制性能评价

6.4.1 BCNN 模型性能评价

为了测试 BCNN 模型对不同强度的 DDoS 攻击事件的监控性能, 本文在流量矩阵数据集中设置了表 6 所示的两类 DDoS 攻击强度分布. 通过调整流量矩阵中 DDoS 报文数量占矩阵中总报文个数的百分比, 反映 DDoS 攻击事件强度从小到大的变化. 在构建恶意的流量矩阵样本时通过随机插入 DDoS 攻击报文, 在流量矩阵中形成一种较为平均的理想化攻击报文分布, 以避免分布不均匀造成特征的不一致.

表 6 DDoS 攻击强度分类

类型	当前网络中DDoS数据包所占比例 (攻击强度)											
单一	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%	11%	12%
区间	1%–3%			4%–7%			8%–12%			13%–17%		

单一强度分布流量矩阵数据集: 包括攻击强度为 1%–12% 的 12 个不同数据集, 其中每个流量矩阵的 DDoS 攻击强度是相同的, 旨在测试 BCNN 模型的准确度性能.

区间强度分布流量矩阵数据集: 本文设置了攻击强度区间为 1%–3%、4%–7%、8%–12%、13%–17% 的 4 个数据集, 每个数据集中的流量矩阵样本包含区间内随机的 DDoS 攻击强度. 在监控场景中, 即使某时刻网络中 DDoS 攻击的强度稳定, 但在抓取流量矩阵时, 连续矩阵样本包含的 DDoS 强度仍然会出现小幅波动. 因此本文的

区间数据集旨在验证 BCNN 模型面对强度震荡的矩阵样本时的性能表现, 同时对比验证单一数据集上的性能表现, 预期应该相近于区间内包含的几个强度上准确度的平均水平.

在训练中, 我们设置训练 epochs=100, batchsize=64, 学习率从 0.01 开始, 基于 ReduceLROnPlateau 方法以 0.1 为调整倍数进行动态衰减. 表 7 中我们分别在训练集、验证集和测试集测试了 BCNN 模型的性能表现.

表 7 BCNN 模型性能

DDoS	训练集						验证集						测试集					
强度 (%)	Loss	*Acc*	*F1-score*	*Pre*	*Recall*	AUC	Loss	*Acc*	*F1-score*	*Pre*	*Recall*	AUC	Loss	*Acc*	*F1-score*	*Pre*	*Recall*	AUC
1	0.5706	**0.7024**	0.703307	0.708924	0.704696	0.7651	0.5833	**0.685**	0.684298	0.688897	0.686666	0.7651	0.5908	**0.692**	0.689915	0.694125	0.690430	0.7651
2	0.4872	**0.7474**	0.771003	0.775233	0.771763	0.8405	0.4651	**0.771**	0.770354	0.775587	0.77104	0.8405	0.4698	**0.765**	0.764064	0.768263	0.764545	0.8405
3	0.4194	**0.8033**	0.804723	0.814541	0.806141	0.8816	0.4341	**0.803**	0.801709	0.810455	0.80294	0.8817	0.4202	**0.784**	0.779231	0.797944	0.782461	0.8981
4	0.3804	**0.8196**	0.828728	0.830845	0.828893	0.9032	0.357	**0.838**	0.837910	0.839684	0.837512	0.9033	0.3312	**0.843**	0.842965	0.844653	0.843846	0.9033
5	0.2983	**0.8734**	0.870368	0.870781	0.870319	0.94	0.2779	**0.883**	0.883442	0.883486	0.883497	0.9401	0.3018	**0.872**	0.871995	0.872605	0.872391	0.9401
6	0.2495	**0.8959**	0.898117	0.898487	0.898122	0.9493	0.2416	**0.888**	0.888419	0.888469	0.888854	0.9494	0.2275	**0.909**	0.908744	0.908376	0.909333	0.9494
7	0.2004	**0.911**	0.914567	0.914558	0.914602	0.9694	0.207	**0.908**	0.90785	0.907678	0.908171	0.9694	0.1904	**0.913**	0.912736	0.912531	0.912987	0.9694
8	0.2102	**0.9164**	0.921995	0.922004	0.921988	0.9632	0.2052	**0.92**	0.919456	0.919449	0.919467	0.9632	0.2125	**0.919**	0.918971	0.918988	0.918956	0.9633
9	0.1892	**0.9254**	0.925418	0.925621	0.925411	0.9688	0.1509	**0.944**	0.943467	0.943515	0.943715	0.9689	0.2047	**0.918**	0.917984	0.918493	0.918071	0.9689
10	0.1065	**0.9557**	0.961571	0.961627	0.961702	0.9873	0.0947	**0.966**	0.965469	0.965663	0.965383	0.9873	0.0936	**0.964**	0.963842	0.964397	0.963413	0.9873
11	0.1065	**0.9596**	0.968709	0.969071	0.968718	0.9857	0.0997	**0.961**	0.960976	0.961301	0.961029	0.9858	0.1018	**0.958**	0.957986	0.958594	0.958	0.9858
12	0.034	**0.987**	0.989142	0.989118	0.989181	0.993	0.0342	**0.99**	0.989491	0.989693	0.989399	0.993	0.0309	**0.989**	0.988962	0.98902	0.988905	0.9930
1–3	0.4379	**0.7866**	0.785966	0.786015	0.78595	0.8604	0.475	**0.768**	0.768309	0.768447	0.768282	0.8605	0.4832	**0.77**	0.76982	0.76986	0.770526	0.8605
4–7	0.1586	**0.9447**	0.872135	0.896764	0.876594	0.958	0.3061	**0.875**	0.874419	0.897446	0.879594	0.9581	0.3046	**0.881**	0.878586	0.905556	0.878323	0.9580
8–12	0.122	**0.946**	0.945419	0.946752	0.94599	0.977	0.1203	**0.947**	0.946339	0.948296	0.945826	0.9771	0.1294	**0.942**	0.941954	0.94355	0.94206	0.9771
13–17	0.0145	**0.9949**	0.996714	0.996712	0.996715	0.998	0.0087	**0.998**	0.997499	0.997511	0.99749	0.998	0.0112	**0.996**	0.995986	0.995875	0.996106	0.9980

首先, 在单一区间强度上, 当 DDoS 强度从 1% 增加到 12% 时, BCNN 在测试集上可以给出 69.2%–98.9% 的 *Acc*, 而在训练集和验证集上的表现类似. 这证明了我们提出的 BCNN 模型具有良好的泛化能力, 没有出现过拟合, 能够有效识别流量矩阵 DDoS 攻击流量的存在性. 同时, 实验结果可以看出 BCNN 模型在 3% 和 7% 处有两个相对明显的准确度跨度提升. 因此, 在表 6 中区间强度数据集的设计上为了避免这种明显跨度干扰模型的学习, 设计了不完全平均的区间划分. 在图 12 中, 我们选取了 DDoS 强度分别为 1%、3%、5% 和 10% 的 4 个数据集, 详细展示了 BCNN 在训练过程中的 *Acc*、Loss 和 ROC 变化曲线, 以证明随着流量矩阵中 DDoS 攻击事件强度的增加, BCNN 模型监控精度的逐渐提升.

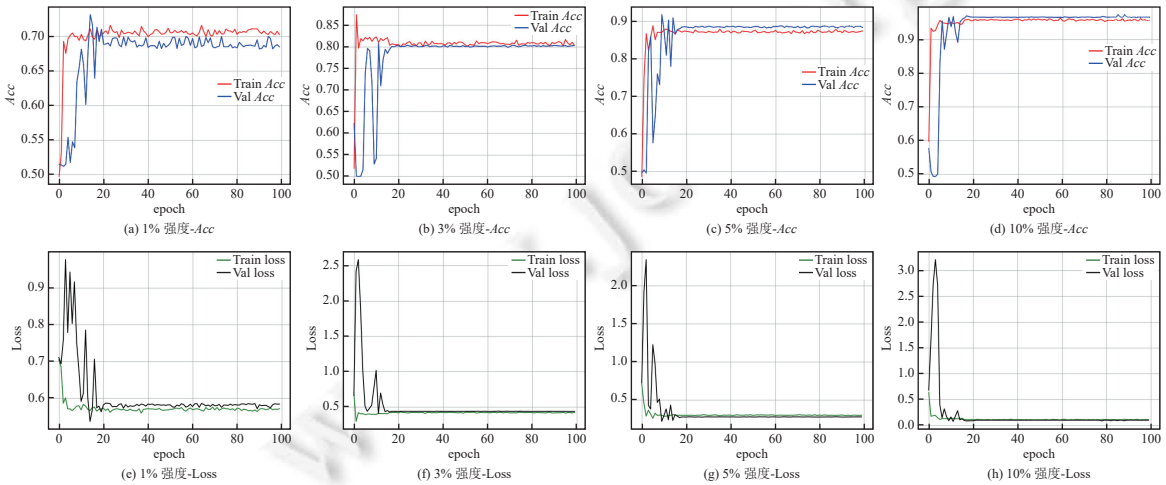
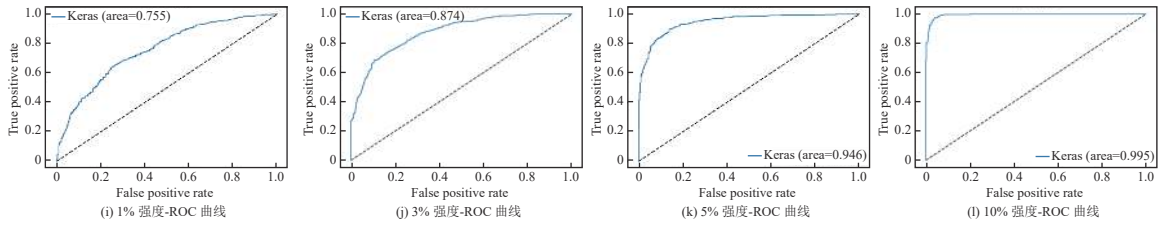


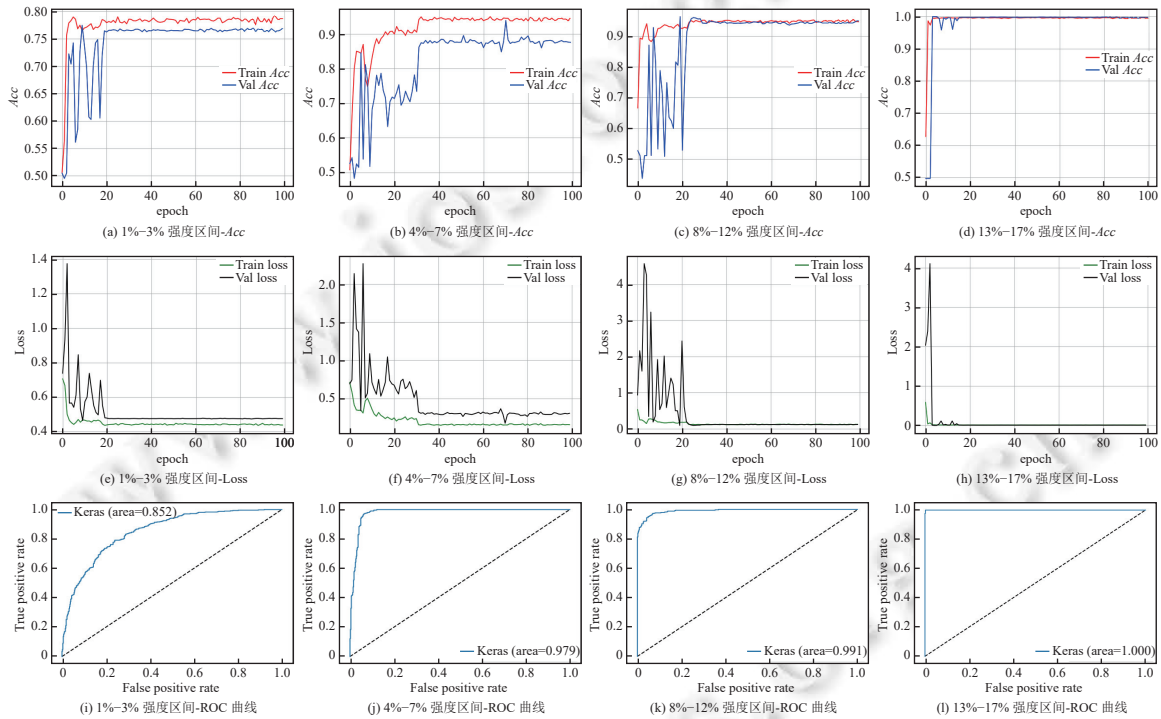
图 12 BCNN 模型在单一强度流量矩阵数据集上的表现



(i) 1% 强度-ROC 曲线 (j) 3% 强度-ROC 曲线 (k) 5% 强度-ROC 曲线 (l) 10% 强度-ROC 曲线

图 12 BCNN 模型在单一强度流量矩阵数据集上的表现 (续)

同样的, 我们观察表 7 中 BCNN 在区间强度数据集流量矩阵时有着良好表现. 在图 13 中, 我们展示了 4 个区间数据集上 BCNN 的 *Acc*、Loss 和 ROC 曲线. 整体而言, BCNN 模型在区间强度上的表现大致等同于区间内单一强度上表现的平均水平, 这符合 BCNN 模型在单一强度数据集上的表现. 当强度区间上升到 13%–17% 时, BCNN 模型的 *Acc* 达到 99.6%, 又因为 BCNN 在 12% 单一强度的 *Acc* 达到 98.9%. 因此可以判断当 DDoS 事件强度超过 12% 之后, BCNN 的 *Acc* 能够近似达到 100%.



(a) 1%–3% 强度区间-*Acc* (b) 4%–7% 强度区间-*Acc* (c) 8%–12% 强度区间-*Acc* (d) 13%–17% 强度区间-*Acc*

(e) 1%–3% 强度区间-Loss (f) 4%–7% 强度区间-Loss (g) 8%–12% 强度区间-Loss (h) 13%–17% 强度区间-Loss

(i) 1%–3% 强度区间-ROC 曲线 (j) 4%–7% 强度区间-ROC 曲线 (k) 8%–12% 强度区间-ROC 曲线 (l) 13%–17% 强度区间-ROC 曲线

图 13 BCNN 模型在区间强度流量矩阵数据集上的表现

6.4.2 性能对比

进一步地, 基于表 7, 我们得到如图 14 所示的由低到高的曲线, 其中横坐标表示网络中 DDoS 攻击事件的强度, 纵坐标是在对应强度测试集上的 *Acc*. 不同颜色的数据柱用于指出 BCNN 模型在不同强度数据集上的性能表现. 同一颜色的数据柱表示该阶段表示出了特征近似的性能趋势. 可以看出, 当攻击强度大于等于 6% 时, BCNN 模型可以给出 90.9% 以上的准确度, 当攻击强度大于等于 10% 时, 能够给出 95.8% 以上的准确度. 在此基础上, 如表 8 所示, 我们对比了近期的 4 个工作在面对不同的 DDoS 攻击强度 (I_*n*) 环境下的识别准确度 (*Acc_n*).

首先, 经典的 DDoS 攻击监控工作大多基于对网络状态变化的观察[14], 例如最新的 Li 等人提出的基于 ϕ 熵的方法[42]. 在实验中, 当 DDoS 强度分别达到 20%、30%、40% 和 50% 时, ϕ 熵的变化幅度分别为 3.2%、6.9%、12.0% 和 18.4%. 如果我们将 10% 的 ϕ 熵的变化视为能够 100% 判断 DDoS 攻击发生的条件. 那么在 6% 和 10% DDoS 强度下, 对 DDoS 攻击行为的识别准确率则仅为 54.8% 和 58%.

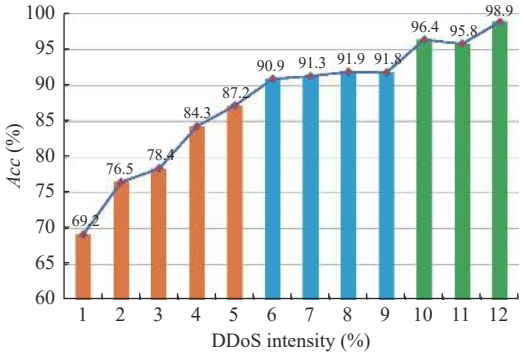


图 14 BCNN 模型 DDoS 事件监控 *Acc* 曲线

表 8 BCNN 性能对比 (%)

算法	I_1	*Acc*_1	I_2	*Acc*_2	I_3	*Acc*_3
Li等人[42]	36	100	6	54.8	10	58
Jog等人[43]	26.5	96	6	52	10	76
Zhou等人[44]	33	94.1	6	58.02	10	63.4
Segura等人[45]	20	99.7	6	84.2	10	88.65
Ours	**12**	**98.9**	**6**	**90.9**	**10**	**96.4**

实际上, 这种仅基于网络状态变化的监控方法无法准确地发现尚处于低强度的 DDoS 行为. 因此, 最近的一些工作将网络状态和流量信息结合用于检测. 比如, 基于回归模型的 Jog 等人提出的方法[43]在 DDoS 强度为 26.5% 时准确度能达到 96%, 而在 6% 和 10% 强度时准确率仅为 52% 和 76%. 同样, Zhou 等人[44]的决策树方法在 DDoS 强度为 33% 时准确率能达到 94.1%, 而在 6% 和 10% 强度时准确率仅为 58% 和 64%. 尽管 Segura 等人[45]提出算法在 20% 的强度时拥有接近 100% 的准确率, 但在 6% 和 10% 的强度下, 准确率仅为 84.2% 和 88.65%. 相比之下, 本文提出的 BCNN 实时监控模型具有明显的性能优势, 不仅在 6% 和 10% 的强度下有着最高的 90.9% 和 96.4% 的准确度. 同时在 DDoS 强度为 12% 时, 就能够达到 98.9% 的近乎 100% 的准确度.

显然, 本文提出的通过监控流量矩阵的方法能够更灵敏且准确地监控到网络中更低强度 DDoS 攻击行为的发生, 从而实现快速响应和防御. 据调研, 在现有的 DDoS 攻击事件监控研究中, 尚无本机制类似的通过识别 DDoS 攻击报文存在性来监控 DDoS 攻击行为的工作, 具有一定创新性.

6.4.3 BCNN 模型开销评价

本文在实验中对提出的 BCNN 模型以及优化前的 Base-CNN 的开销进行了详细对比. 如表 9 所示, 包括参数个数、模型存储大小、内存访问量、FLOPs、计算强度、推理时间和模型性能.

表 9 BCNN 模型开销

开销类型	Base-CNN	BCNN
模型参数量	113 566	11 114
模型大小 (MB)	0.443	0.004
内存访问量 (MB)	1.404	0.943
FLOPs	2.2×10^7	5.3×10^7
计算强度 (FLOPs/B)	14.9	53.6
推理时间 (ms)	0.399	0.262
模型性能 (*Acc*, DDoS强度)	10%, 98.3%	10%, 96.4%

1) 在模型体积上, 与 Base-CNN 模型相比, BCNN 的模型参数量和存储大小压缩了 90.2% 和 99%. 这在于本文不仅使用 GAP 大幅减少全连接参数量, 而且使用二值化卷积核的方法压缩了卷积层参数的存储大小.

2) BCNN 具有更深的卷积结构, 在推理过程中, BCNN 的计算次数为 5.3×10^7 FLOPs, 是 Base-CNN 的 2.4 倍, 但其内存访问量减少了 32.8%, 这使得 BCNN 的计算强度提高了 3.59 倍.

为了对比模型在本文实验使用的 GTX1080 显卡上的运行速度, 图 15 展示了显卡的 roof-line model[46], 并基于模型的计算强度指标标注了两个模型. GTX1080 的最大算力和带宽分别为 9 TFLOP/s 和 382 GB/s. 可以看出, 由于计算强度的不同, Base-CNN 只能使用 显卡 5.5 TFLOP/s 的计算能力, 而 BCNN 可以使用完整的 9 TFLOP/s 计算能力进行推理, 这使得 BCNN 的单张流量矩阵推理时间减少了 34.3%, 仅耗时 0.262 ms. 最重要的是, 相比于

Base-CNN 对 10% 强度 DDoS 攻击行为给出的 98.3% 的识别准确度, BCNN 模型在相同条件下也可以给出相近的 96.4% 的识别准确率.

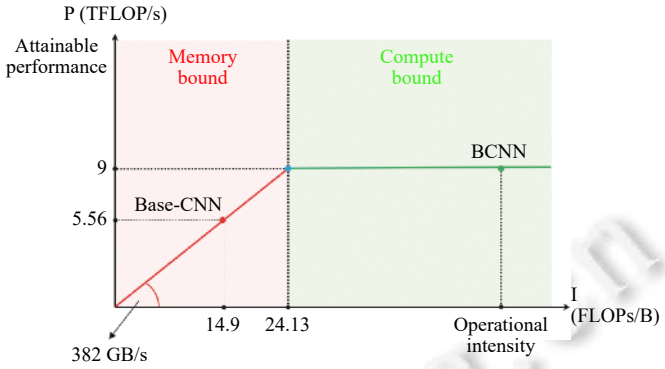


图 15 GTX1080 的 roof-line model

综上所述, 在维持识别性能相近的前提下, BCNN 的轻量化策略极大压缩了模型的参数数量、存储大小和内存访问量, 显著提高了推理速度. 因为没有类似本文使用深度学习模型监控 DDoS 攻击行为的相关工作, 因此, 本文提出的基于 BCNN 的 DDoS 事件监控机制可以视作一种轻量化的方法.

6.5 基于 1DCNN 的 DDoS 流量过滤机制性能

6.5.1 1DCNN 模型性能

在实验中, 我们在单一报文数据集上训练并测试 1DCNN 模型的性能, 设置 epochs=30, batch_size=32, lr=0.1, batch_scale_factor=8, decay=0.001. 为了在训练中达到模型的最佳收敛状态, 设置了从 0.1 开始动态衰减的学习率变化策略 ReduceLROnPlateau. 模型性能如表 10 所示, 其在训练集、验证集和测试集上的 *Acc* 表现分别为 99.95%、99.9% 和 99.43%. 可以看出, 对于本文设计的 1DCNN 模型而言, 在网络报文中区分混杂的 DDoS 攻击报文的性能十分优异, 几乎可以达到 100% 的准确度.

表 10 1DCNN 模型性能

Dataset	Loss	*Acc*	*F*1-*score*	*Pre*	*Recall*	AUC
训练集	0.002 4	**0.999 5**	0.999 500	0.999 497	0.999 502	0.929 7
验证集	0.004 8	**0.999**	0.998 996	0.999 058	0.998 936	0.999 9
测试集	0.018 1	**0.994 3**	0.999 429	0.999 426	0.999 432	0.916 8

6.5.2 性能对比

为了验证 1DCNN 模型的性能优势, 我们对比了现有工作中常用于一维流量检测的 DNN 和 RNN 模型性能. 特别地, 为了验证数据集中地址字段的影响, 避免源或目的地址给出过于明确的区分依据. 我们在原始报文样本基础上删减了 eth.src, eth.dst, ip.src, ip.dst 这 4 部分信息, 形成 1×38 的无地址样本作为对照, 得到表 11 中的性能. 其中 P 代表模型的训练参数个数, LR 代表模型训练完毕后动态衰减的学习率大小, 下标 1 和 2 分别对应模型在有无地址字段的数据集上的不同表现.

表 11 1DCNN 模型性能对比

模型	P_1	$Loss_1$	Acc_1 (%)	LR_1	P_2	$Loss_2$	Acc_2 (%)	LR_2
1DCNN	470	0.004 8	**99.9**	9E–14	470	0.042	**98.4**	9E–9
DNN	1 510	0.044	98.89	9E–14	806	0.190	93.1	9E–14
RNN	506	0.136	96.01	9E–14	506	0.162	95.3	9E–11

横向对比: 我们对比 3 种模型在有无地址字段的两个数据集上的 Loss 和准确度. 可以看出, 地址字段信息的有无体现在准确度上的影响最大值为 5%, 表现在 DNN 模型中, 而在 1DCNN 和 RNN 模型上的影响分别为 1.4% 和 0.7%. 这表明 3 种模型均无法通过源或目的地址等信息获得区分报文的重要特征, 而是基于头部的整体学习检测规律, 验证了本文自建数据集的有效性. 同时, 1DCNN 模型在获得最高准确度的基础上, 有着较为出色的稳定性, 证明其一维卷积核可靠且优秀的分析能力.

纵向对比: 1DCNN 模型能够在更少的参数量时, 实现更高的检测准确度. 在对比中, DNN 模型想获得近似的性能需要 4 倍于 1DCNN 模型的参数数量. 而同样的模型在无地址数据集上, 尽管有着 4 倍的参数个数, DNN 的检测精度仍有着 5% 以上的降低. 这因为 1DCNN 模型的 1 维卷积核在识别中能够提取特征属性之间的空间关联关系, 其特征提取和分析能力明显优于全连接结构. 而对于同样具有关联特征提取能力的 RNN 模型, 本文构建模型参数量近似的 RNN 模型来对比模型的检测精度. 可以看出, 1DCNN 模型在两种数据集上均有着更好的检测精度. 显然, 1DCNN 的空间扩展能力比 RNN 的时序扩展能力更适合 DDoS 检测场景. 尽管参数量相同, 但因为卷积的参数共享能力, 1DCNN 有着更深的模型深度, 特征提取能力更有优势.

6.6 两阶段 DDoS 攻击流量缓解

本文在自建 IPv6-LAN 拓扑上仿真模拟发起 DDoS 攻击, 并执行事件监控、流量识别和过滤实验, 在受害者处监控流量变化如图 16 所示. 不同的是, 在流量过滤阶段, 我们对比了提出的结合 BCNN 和 1DCNN 的两阶段策略和仅使用 1DCNN 的单阶段策略在攻击缓解效率上的差别.

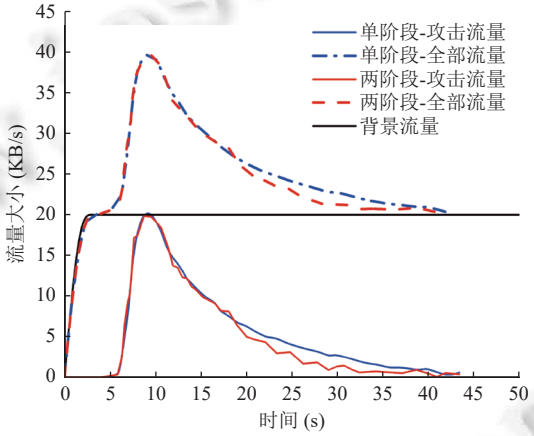


图 16 DDoS 攻击流量变化

在准备阶段, 我们将在东北大学 IPv6 校园网捕获的正常流量使用 Tcpreplay 方法重放形成稳定的 20 KB/s 的背景流量, 即图 16 中 0 s 开始注入的背景流量. 其次, 我们在路由节点_2 上设置流量镜像, 将转发至受害者的全部流量镜像至检测节点进行检测. 接着, 我们在检测节点和路由节点_2 上构建共享数据库, 由检测节点写入检测到的 DDoS 流量源地址信息. 最后, 在路由节点_2 上运行基于 ip6tables 的源地址阻断脚本, 动态读取数据库中的新增攻击者地址阻断转发至受害者以缓解攻击.

从 5 s 开始, 本文在攻击者_1&2 上重放 CIC-DDoS2019 流量集, 基于 NAT 4to6 发起 ICMPv6、TCP 和 UDP 混合 DDoS 攻击, 稳定 DDoS 流量总速率为 20 KB/s. 但与制备实验数据集不同, 为了体现攻击的阻断效果, DDoS 攻击流量的重放没有随机的设置源地址信息, 而是将 DDoS 流量平均分配到 512 个固定的源地址. 前文验证 1DCNN 模型的检测准确度接近 100%, 因此, 我们不考虑误阻断, 将即时 DDoS 流量大小计算为受害者接收总流量减去固定背景流量大小.

观察图 16 可知, DDoS 防御从 10 s 开始, 两种策略在 10–18 s 的过滤速率基本一致, 因为此时两种策略的过滤流程相同, 仅使用 1DCNN 模型, 但在 18 s 后, DDoS 流量的过滤速度出现差异. 此时, DDoS 速率大约为 7.5 KB/s,

占总速率接近 20%, 触发了两阶段策略中的 DDoS 强度阈值判定条件, 即 BCNN 模型启动, IDCNN 模型将以恶意流量矩阵所含报文作为检测源. 对比看出, 从 18 s 开始, 两阶段策略的过滤速度明显高于单阶段策略. 随着 DDoS 强度的下降, 网络中 DDoS 流量占比减小, 单阶段的 1DCNN 模型命中率明显下降. 相比之下, BCNN 模型的引入能够大幅提升攻击流量的命中率, 提升检测效率并降低开销. 同样在 18 s 后也可以看出, 两阶段策略以流量矩阵作为过滤单元, 因此攻击流量的下降呈阶梯状, 且随着 DDoS 强度继续下降, BCNN 对流量矩阵的识别精度也会下降, 连带 1DCNN 无法及时识别到网络中新出现的攻击 IP 发送的报文, 导致 DDoS 流量出现波动. 总体来看, 实验中将 DDoS 攻击速率从 20 KB/s 缓解至约 1 KB/s 时, 单阶段用时 29 s, 而两阶段策略用时 19 s, 减少了 34.5% 的缓解时间.

7 总 结

本文针对 IPv6 网络中的 DDoS 攻击检测问题, 提出了一种基于 BCNN 和 1DCNN 的两阶段 DDoS 攻击防御机制. 本文首先介绍了 DDoS 攻击的相关原理和相关防御工作, 并分析了深度学习模型在流量分析中的优势. 进而, 本文将防御划分为实时监控 DDoS 攻击事件的预检测阶段和快速过滤 DDoS 攻击报文的深度检测阶段, 提出分别使用 BCNN 和 1DCNN 模型作为各阶段的决策核心. 基于此, 在正常网络环境中, 本机制能够在初期快速响应发生的 DDoS 攻击, 尽早介入并过滤 DDoS 攻击报文进行缓解. 本文在 CERNET2 上自建 IPv6-LAN 实验拓扑并引入 CIC-DDoS2019 公开流量集进行实验验证. 经过对比, 本文提出使用的深度学习决策核心在轻量化的同时能够给出更好的性能表现, BCNN 模型针对 1%、5% 和 10% 强度的 DDoS 攻击能够分别达到 69.2%、87.2% 和 96.4% 的监控成功率, 同时 1DCNN 对 DDoS 攻击报文的识别准确率能够达到 99.4%, 结合提出的两阶段防御策略能够有效提升 DDoS 攻击的流量缓解效率.

References:

[1] Engströma V, Lagerströma R. Two decades of cyberattack simulations: A systematic literature review. Computers & Security, 2022, 116: 102681. [doi: 10.1016/j.cose.2022.102681]

[2] Balarezo JF, Wang S, Chavez KG, Al-Hourani A, Kandeepan S. A survey on DoS/DDoS attacks mathematical modelling for traditional, SDN and virtual networks. Engineering Science and Technology, an Int'l Journal, 2022, 31: 101065. [doi: 10.1016/j.jestch.2021.09.011]

[3] Google IPv6 Statistics. 2022. https://www.google.com/intl/en/ipv6/statistics.html

[4] Liu N, Xia J, Cai ZP, Yang T, Hou BN, Wang ZL. A survey on IPv6 security threats and defense mechanisms. In: Proc. of the 8th Int'l Conf. on Adaptive and Intelligent Systems. Qinghai: Springer, 2022: 583–598. [doi: 10.1007/978-3-031-06794-5_47]

[5] Tayyab M, Belaton B, Anbar M. ICMPv6-based DoS and DDoS attacks detection using machine learning techniques, open challenges, and blockchain applicability: A review. IEEE Access, 2020, 8: 170529–170547. [doi: 10.1109/ACCESS.2020.3022963]

[6] Li YB, Yang W, Zhou Z, Liu QY, Li Z, Li S. P4-NSAF: Defending IPv6 networks against ICMPv6 DoS and DDoS attacks with P4. In: Proc. of the 2022 IEEE Int'l Conf. on Communications (ICC). Seoul: IEEE, 2022. 5005–5010. [doi: 10.1109/ICC45855.2022.9839137]

[7] Elejla OE, Anbar M, Hamouda S, Faisal S, Bahashwan AA, Hasbullah IH. Deep-learning-based approach to detect ICMPv6 flooding DDoS attacks on IPv6 networks. Applied Sciences, 2022, 12(12): 6150. [doi: 10.3390/app12126150]

[8] Bahashwan AA, Anbar M, Hanshi SM. Overview of IPv6 based DDoS and DoS attacks detection mechanisms. In: Proc. of the 1st Int'l Conf. on Advances in Cyber Security. Penang: Springer, 2019. 153–167. [doi: 10.1007/978-981-15-2693-0_11]

[9] Zhao JJ, Jing XY, Yan Z, Pedrycz W. Network traffic classification for data fusion: A survey. Information Fusion, 2021, 72: 22–47. [doi: 10.1016/j.inffus.2021.02.009]

[10] Wu JP, Wang JH, Yang JH. CNGI-CERNET2: An IPv6 deployment in China. ACM SIGCOMM Computer Communication Review, 2011, 41(2): 48–52. [doi: 10.1145/1971162.1971170]

[11] Sharafaldin I, Lashkari AH, Hakak S, Ghorbani AA. Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In: Proc. of the 2019 Int'l Carnahan Conf. on Security Technology (ICCST). Chennai: IEEE, 2019. 1–8. [doi: 10.1109/CCST.2019.8888419]

[12] Alsadhan A, Hussain A, Liatsis P, Alani M, Tawfik H, Kendrick P, Francis H. Locally weighted classifiers for detection of neighbor discovery protocol distributed denial-of-service and replayed attacks. Trans. on Emerging Telecommunications Technologies, 2022, 33(3): e3700. [doi: 10.1002/ett.3700]

[13] Novo O. Making constrained things reachable: A secure IP-agnostic NAT traversal approach for IoT. ACM Trans. on Internet Technology, 2018, 19(1): 3. [doi: 10.1145/3230640]

[14] Yuan J, Mills K. Monitoring the macroscopic effect of DDoS flooding attacks. IEEE Trans. on Dependable and Secure Computing, 2005, 2(4): 324–335. [doi: 10.1109/TDSC.2005.50]

[15] Singh K, Dhindsa KS, Bhushan B. Threshold-based distributed DDoS attack detection in ISP networks. Turkish Journal of Electrical Engineering & Computer Sciences, 2018, 26(4): 1796–1811. [doi: 10.3906/elk-1712-3]

[16] Liu CH, Yeh YT. Monitoring DDoS by using SDN. Journal of Internet Technology, 2016, 17(2): 341–348.

[17] Liu ZX, Manousis A, Vorsanger G, Sekar V, Braverman V. One sketch to rule them all: Rethinking network flow monitoring with UnivMon. In: Proc. of the 2016 ACM SIGCOMM Conf. Florianopolis: ACM, 2016. 101–114. [doi: 10.1145/2934872.2934906]

[18] Wang B, Zheng Y, Lou WJ, Hou YT. DDoS attack protection in the era of cloud computing and software-defined networking. Computer Networks, 2015, 81: 308–319. [doi: 10.1016/j.comnet.2015.02.026]

[19] Galeano-Brajones J, Carmona-Murillo J, Valenzuela-Valdés JF, Luna-Valero F. Detection and mitigation of DoS and DDoS attacks in IoT-based Stateful SDN: An experimental approach. Sensors, 2020, 20(3): 816. [doi: 10.3390/s20030816]

[20] Xie Y, Yu SZ. Monitoring the application-layer DDoS attacks for popular websites. IEEE/ACM Trans. on Networking, 2009, 17(1): 15–25. [doi: 10.1109/TNET.2008.925628]

[21] Biswas R, Kim S, Wu J. Sampling rate distribution for flow monitoring and DDoS detection in datacenter. IEEE Trans. on Information Forensics and Security, 2021, 16: 2524–2534. [doi: 10.1109/TIFS.2021.3054522]

[22] Baskar M, Ramkumar J, Karthikeyan C, Anbarasu V, Balaji A, Arulananth TS. Low rate DDoS mitigation using real-time multi threshold traffic monitoring system. Journal of Ambient Intelligence and Humanized Computing, 2021: 1–9. [doi: 10.1007/s12652-020-02744-y]

[23] Zaib MH, Bashir F, Qureshi KN, Kausar S, Rizwan M, Jeon G. Deep learning based cyber bullying early detection using distributed denial of service flow. Multimedia Systems, 2022, 28(6): 1905–1924. [doi: 10.1007/s00530-021-00771-z]

[24] Yuan XY, Li CH, Li XL. DeepDefense: Identifying DDoS attack via deep learning. In: Proc. of the 2017 IEEE Int'l Conf. on Smart Computing (SMARTCOMP). Hong Kong: IEEE, 2017. 1–8. [doi: 10.1109/SMARTCOMP.2017.7946998]

[25] Rehman SU, Khaliq M, Imtiaz SI, Rasool A, Shafiq M, Javed AR, Jalil Z, Bashir AK. DIDDOS: An approach for detection and identification of distributed denial of service (DDoS) cyberattacks using gated recurrent units (GRU). Future Generation Computer Systems, 2021, 118: 453–466. [doi: 10.1016/j.future.2021.01.022]

[26] Ali SHA, Ozawa S, Ban T, Nakazato J, Shimamura J. A neural network model for detecting DDoS attacks using darknet traffic features. In: Proc. of the 2016 Int'l Joint Conf. on Neural Networks (IJCNN). Vancouver: IEEE, 2016. 2979–2985. [doi: 10.1109/IJCNN.2016.7727577]

[27] Saad RMA, Anbar M, Manickam S, Alomari E. An intelligent ICMPv6 DDoS flooding-attack detection framework (v6IIDS) using back-propagation neural network. IETE Technical Review, 2016, 33(3): 244–255. [doi: 10.1080/02564602.2015.1098576]

[28] Ye J, Cheng XY, Zhu J, Feng LT, Song L. A DDoS attack detection method based on SVM in software defined network. Security and Communication Networks, 2018, 2018: 9804061. [doi: 10.1155/2018/9804061]

[29] Parra GDLT, Rad P, Choo KKR, Beebe N. Detecting Internet of Things attacks using distributed deep learning. Journal of Network and Computer Applications, 2020, 163: 102662. [doi: 10.1016/j.jnca.2020.102662]

[30] Premkumar M, Sundararajan TVP. DLDM: Deep learning-based defense mechanism for denial of service attacks in wireless sensor networks. Microprocessors and Microsystems, 2020, 79: 103278. [doi: 10.1016/j.micpro.2020.103278]

[31] Doriguzzi-Corin R, Millar S, Scott-Hayward S, Martinez-Del-Rincon J, Siracusa D. LUCID: A practical, lightweight deep learning solution for DDoS attack detection. IEEE Trans. on Network and Service Management, 2020, 17(2): 876–889. [doi: 10.1109/TNSM.2020.2971776]

[32] Asad M, Asim M, Javed T, Beg MO, Mujtaba H, Abbas S. DeepDetect: Detection of distributed denial of service attacks using deep learning. The Computer Journal, 2020, 63(7): 983–994. [doi: 10.1093/comjnl/bxz064]

[33] Hwang RH, Peng MC, Huang CW, Lin PC, Nguyen VL. An unsupervised deep learning model for early network traffic anomaly detection. IEEE Access, 2020, 8: 30387–30399. [doi: 10.1109/ACCESS.2020.2973023]

[34] Cil AE, Yildiz K, Buldu A. Detection of DDoS attacks with feed forward based deep neural network model. Expert Systems with Applications, 2021, 169: 114520. [doi: 10.1016/j.eswa.2020.114520]

[35] Wei GL, Wang ZH. Adoption and realization of deep learning in network traffic anomaly detection device design. Soft Computing, 2021, 25(2): 1147–1158. [doi: 10.1007/s00500-020-05210-1]

[36] Cao Y, Gao Y, Tan RJ, Han QB, Liu ZT. Understanding Internet DDoS mitigation from academic and industrial perspectives. IEEE Access, 2018, 6: 66641–66648. [doi: 10.1109/ACCESS.2018.2877710]

[37] Shahraki A, Abbasi M, Taherkordi A, Jurcut AD. Active learning for network traffic classification: A technical study. IEEE Trans. on Cognitive Communications and Networking, 2021, 8(1): 422–439. [doi: 10.1109/TCCN.2021.3119062]

[38] Maswood MMS, Mamun MMI, Huang DJ, Medhi D. A sliding window based monitoring scheme to detect and prevent DDoS attack in data center networks in a dynamic traffic environment. In: Proc. of the 39th IEEE Sarnoff Symp. Newark: IEEE, 2018. 1–6. [doi: 10.1109/SARNOF.2018.8720399]

[39] Zhuang L, Qi HY, Wang TG, Zhang ZJ. A deep-learning-powered near-real-time detection of railway track major components: A two-stage computer-vision-based method. IEEE Internet of Things Journal, 2022, 9(19): 18806–18816. [doi: 10.1109/JIOT.2022.3162295]

[40] Simons T, Lee DJ. A review of binarized neural networks. Electronics, 2019, 8(6): 661. [doi: 10.3390/electronics8060661]

[41] Courbariaux M, Hubara I, Soudry D, El-Yaniv R, Bengio Y. Binarized neural networks: Training deep neural networks with weights and activations constrained to +1 or −1. arXiv:1602.02830, 2016.

[42] Li RY, Wu B. Early detection of DDoS based on ϕ-entropy in SDN networks. In: Proc. of the 4th IEEE Information Technology, Networking, Electronic and Automation Control Conf. (ITNEC). Chongqing: IEEE, 2020. 731–735. [doi: 10.1109/ITNEC48623.2020.9084885]

[43] Jog M, Natu M, Shelke S. Distributed and predictive-preventive defense against DDoS attacks. In: Proc. of the 16th Int'l Conf. on Distributed Computing and Networking. Goa: ACM, 2015. 29. [doi: 10.1145/2684464.2684503]

[44] Zhou BJ, Li J, Ji YS, Guizani M. Online internet traffic monitoring and DDoS attack detection using Big Data frameworks. In: Proc. of the 14th Int'l Wireless Communications & Mobile Computing Conf. (IWCMC). Limassol: IEEE, 2018. 1507–1512. [doi: 10.1109/IWCMC.2018.8450335]

[45] Segura GAN, Skaperas S, Chorti A, Mamatas L, Margi CB. Denial of service attacks detection in software-defined wireless sensor networks. In: Proc. of the 2020 IEEE Int'l Conf. on Communications Workshops (ICC Workshops). Dublin: IEEE, 2020. 1–7. [doi: 10.1109/ICCWorkshops49005.2020.9145136]

[46] Vitali E, Ficarelli F, Bisson M, Gadioli D, Fatica M, Beccari AR, Palermo G. GPU-optimized approaches to molecular docking-based virtual screening in drug discovery: A comparative analysis. arXiv:2209.05069, 2022.

王郁夫(1995－), 男, 博士生, 主要研究领域为网络安全, DDoS 防御, 人工智能.

易波(1988－), 男, 博士, 讲师, CCF 专业会员, 主要研究领域为软件定义网络, 网络功能虚拟化.

王兴伟(1968－), 男, 博士, 教授, 博士生导师, CCF 杰出会员, 主要研究领域为互联网, 云计算, 网络安全.

黄敏(1968－), 女, 博士, 教授, 博士生导师, 主要研究领域为数据解析与机器学习, 计算智能软件工程.