面向主机入侵检测的多视图对抗攻击防御方法

王飞[1]，钱可涵[3]，吕明琪[2]，朱添田[3]，陈鸿龙[1]
（1.中国石油大学（华东）控制科学与工程学院，山东 青岛 266580；2.浙江工业大学地理信息学院，浙江 湖州 313299；
3.浙江工业大学计算机科学与技术学院，浙江 杭州 310023）

摘　要：主机入侵检测（HID）旨在通过分析主机日志识别攻击行为。针对图神经网络模型在主机入侵检测中易受对抗攻击的问题，提出一种多视图对抗防御方法。通过构建结构与行为双视图以融合多维特征，筛选低迁移性互补模型对，并设计分级投票机制集成异构模型决策，从而提升检测鲁棒性。基于真实的主机内核日志数据集对该方法进行了评测，实验结果表明，该方法优于现有的对抗攻击防御方法，在典型对抗攻击下的恶意节点召回率达到80%以上，较现有单模型防御方法提升约23%，且误报率控制在10%以内，验证了基于迁移性分析的融合策略对增强鲁棒性的有效性。

关键词：对抗攻击；主机入侵检测；溯源图；多模型集成
中图分类号：TP18；TP309
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2025140

Multi-view adversarial attack defending method for host intrusion detection

WANG Fei[1], QIAN Kehan[3], LYU Mingqi[2], ZHU Tiantian[3], CHEN Honglong[1]
1. College of Control Science and Engineering, China University of Petroleum (East China), Qingdao 266580, China
2. School of Geographic Information, Zhejiang University of Technology, Huzhou 313299,China
3. College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China

Abstract: Host-based intrusion detection (HID) aims to identify attack behaviors through the analysis of host logs. Aiming at the problem that the graph neural network model was vulnerable to adversarial attacks in host intrusion detecti on, a multi-view adversarial defense method was proposed. By constructing dual views of structure and behavior to integrate multi-dimensional features, screening low-transferability complementary model pairs, and designing a hierarchical voting mechanism to integrate heterogeneous model decisions, the robustness of detection was enhanced. The efficacy of the proposed method was evaluated using authentic host kernel log datasets. The experimental results demonstrate that the method exhibits superior performance compared to existing adversarial attack defense methods. Specifically, a malicious node recall rate exceeding 80% is achieved under typical adversarial attacks, representing a 23% increase over existing single-model defense methods. Additionally, the false alarm rate is maintained below 10%, substantiating the efficacy of the transferability analysis-based fusion strategy for robustness enhancement.

Keywords: adversarial attack, host intrusion detection, provenance graph, multi-model ensemble

收稿日期：2025–06–06；**修回日期：**2025–08–08
通信作者：吕明琪，mingqilv@zjut.edu.cn
基金项目：国家自然科学基金资助项目（No.62372410, No.62002324）；浙江省自然科学基金资助项目（No.LZ23F020011）；杭州市重点研发计划基金资助项目（No.2024SZD1A11）；山东省泰山学者青年专家基金资助项目（No.tsqn202312133）；山东省优秀青年科学基金资助项目（No.ZR2022YQ61）
Foundation Items: The National Natural Science Foundation of China (No.62372410, No.62002324), The Natural Science Foundation of Zhejiang Province (No. LZ23F020011), The Key Research and Development Program of Hangzhou (No. 2024SZD1A11), The Shandong Provincial Taishan Scholar Program (No. tsqn202312133), The Shandong Provincial Natural Science Foundation (No.ZR2022YQ61)

0 引言

主机入侵检测旨在通过持续收集和分析主机（如服务器、边缘设备）的日志数据来发现针对主机的攻击行为[1]。近年来，高级持续性威胁（APT, advanced persistent threat）等针对主机的攻击日益复杂。攻击者通过动态调整攻击策略，并通过多种攻击技术绕过防御系统，试图长期潜伏于主机中而不触发告警[2]。为了有效检测此类复杂攻击，先进的主机入侵检测系统采用底层的内核日志数据作为检测对象。内核日志数据包含操作系统底层的系统事件，并可基于系统事件的上下文关联表征为一个有向无环图，称为溯源图。其中，溯源图的每个节点代表一个系统实体，每条边代表一个系统事件。溯源图能够捕捉系统内的细粒度上下文关联，因此对多步、长期的复杂攻击具有较好的检测效果[3]。

由于溯源图的结构和语义复杂庞大，近期研究利用深度学习技术分析溯源图，并建立主机入侵检测模型[4-5]。鉴于溯源图的特殊数据结构，图神经网络（GNN, graph neural network）是应用最广泛的深度学习技术。具体来说，主机入侵检测被定义为一个节点分类任务，首先利用图嵌入技术将溯源图的每个节点编码成一个语义化特征向量，然后训练节点分类器来检测表示攻击事件的恶意节点。

虽然GNN在主机入侵检测任务中取得了良好性能，但最近的研究表明，GNN内生的鲁棒性缺陷导致其易受对抗攻击的影响[6-7]。在主机入侵检测场景中，攻击者可通过修改攻击行为来逃逸GNN的检测[8-9]。例如，删除节点/边以掩盖明显的恶意活动，添加节点/边以模仿良性活动。在高度对抗的网络空间环境中，主机入侵检测模型的鲁棒性缺陷会导致严重的安全隐患。

在实际应用场景中，攻击者通常只能实施黑盒对抗攻击，即攻击者无法获得目标主机入侵检测模型的架构、参数、训练数据等内部细节，而只能观测目标主机入侵检测模型的输入输出。黑盒对抗攻击的具体步骤为首先攻击者通过查询目标主机入侵检测模型的输入输出来创建一个模仿其决策行为的代理模型，然后在代理模型上生成对抗样本并迁移到目标主机入侵检测模型上。因此，对抗样本的迁移性是黑盒对抗攻击能够成功的重要前提。

为此，本文在探索不同基于溯源图的主机入侵检测模型间的对抗迁移性的基础上，提出了一种面向主机入侵检测的多视图对抗攻击防御方法。首先，通过设计不同的溯源图节点初始特征以及采用不同的底座GNN模型，创建多个主机入侵检测模型。其次，针对多种黑盒对抗攻击算法，测试对抗攻击样本在不同主机入侵检测模型间的迁移性。最后，根据对抗迁移性集成多种主机入侵检测模型，构建防御系统。

1 问题的提出

1.1 溯源图概念

溯源图的数据来自操作系统内核日志，即大量的系统事件，每个系统事件表示为一个三元组（主体、操作、客体），代表主体对客体执行操作。其中，主体和客体为系统实体（如进程、文件、注册表、套接字），表1总结了本文考虑的操作类型。溯源图可通过将指向同一个系统实体的客体和主体合并为一个节点来构建。溯源图可捕捉系统实体之间的上下文关联，便于对时间上相距甚远的系统实体进行推理，因此在追踪隐蔽和持久的主机攻击方面非常有效[10]。

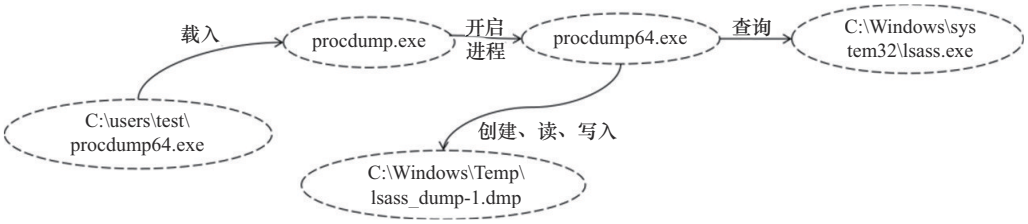
表1 主体、客体和操作类型

主体类型	客体类型	操作类型
进程	进程	开启、结束
进程	文件	创建、关闭、删除、写入、读、清理、遍历、唤醒
进程	注册表	打开、关闭、创建、遍历、查询、删除、设置、刷新
进程	套接字	接受连接、断开连接、发送、接收、拷贝、连接、重新连接、重传
文件	进程	载入

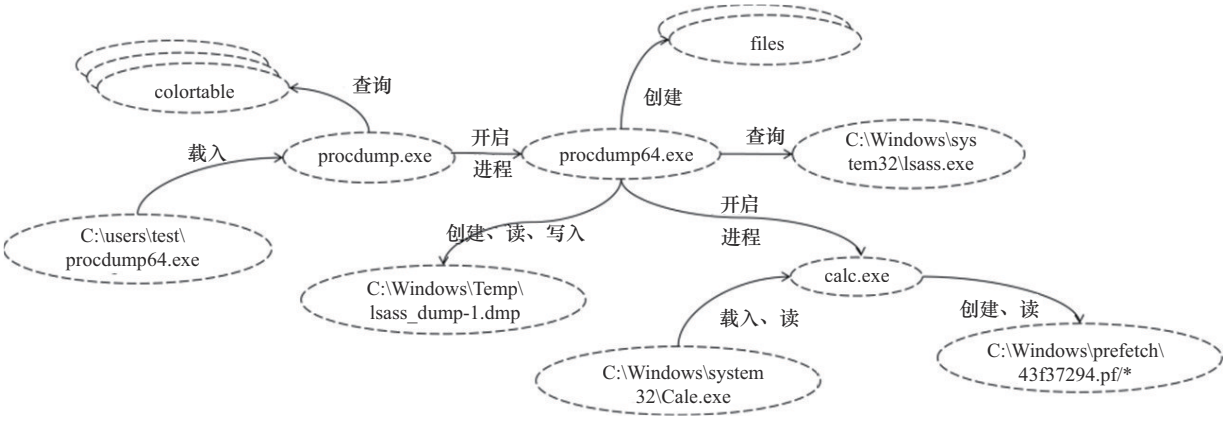
1.2 问题案例

图1(a)给出了一个操作系统凭证转储攻击（对应ATT&CK的攻击技术T1003.001）的简化案例。首先，攻击者将恶意载荷写入磁盘并创建进程“procdump”，该进程随后创建一个新的进程“procdump64”。其次，进程“procdump64”读取存储在本地安全授权子系统服务（LSASS）中的凭证数据，并将其转储到本地文件“lsass_dump-1.dmp”中。最后，攻击者可以使用mimikatz等工具从本地文件中读取凭证明文或散列密码。

主机入侵检测模型可通过学习恶意进程节点



(a) 操作系统凭证转储攻击对应溯源图（简化版）



(b) 经对抗攻击后的溯源图

图 1　主机入侵检测模型对抗攻击案例

（如“procdump”“procdump64”）的行为特征来检测它们（具体见 2.1 节）。例如，主机入侵检测模型可发现异常的子图结构（如频繁操作注册表和文件）或异常的系统事件（如访问敏感的系统文件）。

然而，攻击者可以通过修改他们的行为来绕过主机入侵检测模型。例如，如图 1(b)所示，攻击者试图通过创建额外合法进程和操作额外文件来干扰主机入侵检测模型。由于本文关注的是基于深度学习的主机入侵检测模型，利用这类模型对对抗样本的脆弱性，可以通过对抗攻击技术对输入进行细微扰动，从而改变模型的检测结果，实现规避入侵检测的目的[11-12]。

攻击者可以不断检索目标模型以生成有效的对抗攻击，因此防御对抗攻击十分困难。然而，本文通过实验发现能逃逸一种主机入侵检测模型的对抗样本不一定能逃逸另一种主机入侵检测模型。例如，假设主机入侵检测模型 A 主要关注操作语义的异常程度，主机入侵检测模型 B 主要关注子图结构的异常程度，则攻击者可通过添加大量合法操作来逃逸入侵检测模型 A 的检测，但此类扰动不会改变子图结构的异常程度，因此可能无法逃逸入侵检测模型 B 的检测。

1.3　威胁模型

本文假设攻击者只能实施黑盒对抗攻击，即攻击者无法获得目标主机入侵检测模型的网络架构、参数、训练数据等内部细节，而只能通过观测输入和输出来与目标主机入侵检测模型交互。因此本文针对攻击者只能实施黑盒对抗攻击的场景[12]。在观测目标主机入侵检测模型的输入和输出的基础上，攻击者可利用各种对抗攻击算法（如生成对抗网络、进化算法、强化学习）来生成对抗攻击样本。此外，由于对抗攻击样本都在特征域生成，攻击者需要在问题域中修改原有行为以使得新的溯源图符合新的特征向量[13]。虽然这是一项极具挑战的任务，但本文假设攻击者总能完成该任务。

此外，与现有工作类似[8-9]，本文假设操作系统、内核日志采集器、主机入侵检测模型等模块是正常工作的，本文仅关注对抗攻击样本是否能逃逸主机入侵检测模型，以及本文方法是否能成功拦截对抗攻击样本。

1.4　方法框架

如图 2 所示，本文方法包含 2 个模块：多视图模型构建模块和多模型集成防御模块。

多视图模型构建模块：利用多种策略来训练多样化的主机入侵检测模型，包括使用不同特征来表

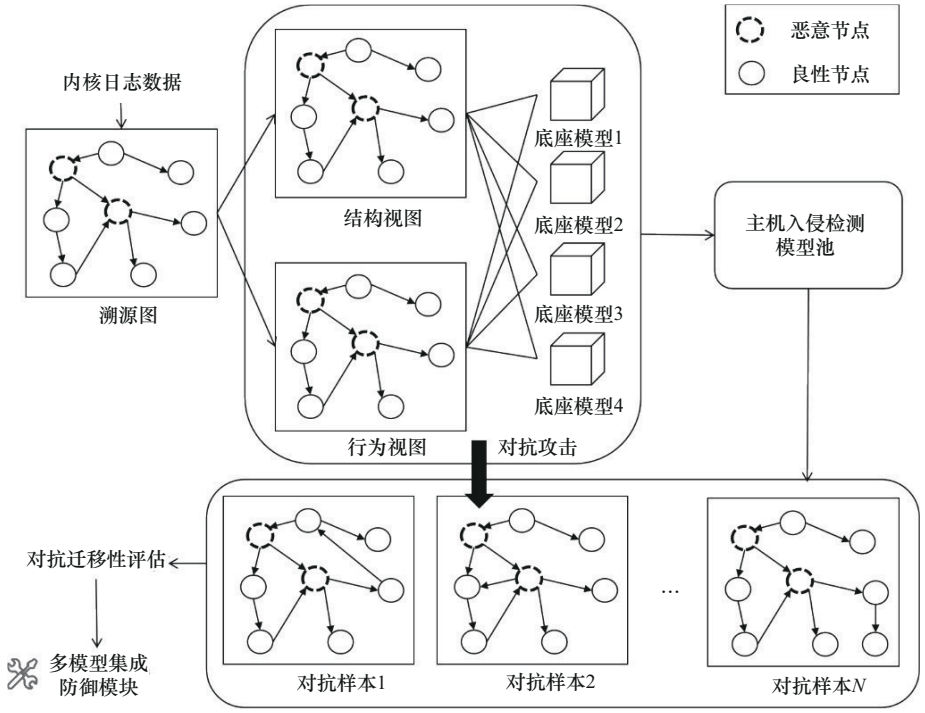


图2　本文方法架构

征溯源图节点和使用不同的底座GNN模型，期望这些多样化的主机入侵检测模型之间的对抗迁移性不一定很高。

多模型集成防御模块：首先，利用多种对抗攻击算法生成的对抗攻击样本来测试多样化的主机入侵检测模型之间的迁移性；然后，通过基于对抗迁移性设计的多种融合策略对多个主机入侵检测模型进行集成，构建对抗攻击防御系统。

2　多视图模型构建

攻击者常利用行为混淆等技术规避单一维度的检测机制，导致仅依赖孤立特征分析会造成大量威胁漏检。为此，本节构建了双重特征表征体系，旨在突破单一视图的语义局限，通过融合异构特征空间中的互补信息来增强模型在对抗环境下的鲁棒性。具体而言，本节将分别构建结构视图和行为视图的节点特征：结构视图通过建模拓扑关系来刻画节点间的交互模式；行为视图则基于敏感活动分析来量化节点的操作语义。本文将主机入侵检测任务定义为一个溯源图节点分类问题。即给定一个溯源图$\mathrm{PG}=(\mathrm{PV},\mathrm{PE})$，将PV中的每个节点分类为良性或恶意，通常分为特征抽取和模型学习2个步骤。

2.1　特征抽取

不同的溯源图节点特征反映了模型关注的不同语义信息，本节从结构视图和行为视图2个角度抽取溯源图节点的特征。

结构视图：与良性节点相比，恶意节点通常与其相邻节点具有不同的交互模式，从而形成不同的局部子图结构[10]。为此，结构视图通过观测溯源图节点周围边类型的分布来形成特征向量。具体来说，首先将所有溯源图节点分为4种类型（包括进程、文件、注册表和套接字），不同类型节点之间的交互可形成27种类型的边，如表1所示。然后，对于每个溯源图节点v_i，创建一个54维向量$\mathbf{se}_i=\left[a_1,a_2,\cdots,a_{21},a_{22},a_{23},\cdots,a_{54}\right]$，作为$v_i$的结构特征向量，其中$a_i(1\leqslant i\leqslant 27)$是$v_i$的第$i$种类型的入边数量，$a_j(28\leqslant j\leqslant 54)$是$v_i$的第$(j-27)$种类型的出边数量。

行为视图：根据文献[13-15]的经验，恶意节点很可能实施敏感操作。例如，数据渗出攻击很有可能会访问敏感文件和远程服务器。为此，本文参考文献[16]提出的敏感操作列表，定义了4个系统实体的34种敏感操作。然后，对于溯源图中的每个节点v_i，创建一个34维向量$\mathbf{be}_i=\left[b_1,b_2,\cdots,b_{21},b_{22},b_{23},\cdots,b_{34}\right]$作为$v_i$的行为特征向量，

其中 b_i 是 v_i 执行第 i 类敏感操作的次数。如果第 i 类敏感操作是二元事件，则 b_i= 1 或 0。

2.2　模型学习

2.1 节中抽取的溯源图节点特征向量只能反映溯源图节点的一阶上下文关联。复杂的主机攻击（如 APT）通常会采用多种技战术、通过多个步骤来完成攻击行为[16]，导致一阶上下文关联难以捕捉行为链条更长的攻击行为。为此，可使用 GNN 模型学习溯源图节点的更高阶上下文关联[4-5,17]。具体来说，本文采用了 4 种有代表性的 GNN 作为底座模型，即图卷积网络（GCN, graph convolutional network）[18]、图采样与聚合（GraphSAGE, graph sample and aggregate）[19]、图注意力网络（GAT, graph attention network）[20] 和图同构网络（GIN, graph isomorphism network）[21]。最终，通过组合 2 种特征视图和 4 种底座模型，可以得到 8 个主机入侵检测模型。

3　多模型集成防御

3.1　对抗攻击

为测试不同主机入侵检测模型间的对抗迁移性，需要生成大量的对抗攻击样本。为保证对抗攻击样本的多样性以实现更全面的对抗迁移性测试，采用多种不同的对抗攻击算法。此外，与普通的针对 GNN 的对抗攻击不同，针对主机入侵检测模型的对抗攻击还需要满足一些限制条件。

3.1.1　主机入侵检测模型对抗攻击场景

针对 GNN 的对抗攻击场景，设 $\boldsymbol{G}=(\boldsymbol{A},\boldsymbol{X})$ 是一个图，其中 $\boldsymbol{A}$ 是表示 $\boldsymbol{G}$ 拓扑结构的邻接矩阵，$\boldsymbol{X}$ 表示节点特征，对抗攻击的目标是在 $\boldsymbol{G}$ 上添加小的扰动，得到一个扰动图 $\boldsymbol{G'}=(\boldsymbol{A'},\boldsymbol{X'})$，使得对 $\boldsymbol{G}$ 的节点分类性能下降。对 $\boldsymbol{A}$ 的扰动称为结构攻击，对 $\boldsymbol{X}$ 的扰动称为特征攻击。在黑盒对抗攻击场景中，攻击者只知道输入溯源图的结构，而不知道目标主机入侵检测模型采用何种特征，因此本文只关注结构攻击。针对主机入侵检测模型的对抗攻击 AK 的形式化定义如式(1)所示，其中，$\boldsymbol{A}$ 是输入溯源图 $\boldsymbol{G}$ 的邻接矩阵，δ 是扰动，目标是通过修改 $\boldsymbol{G}$ 的结构，最大限度增加主机入侵检测模型对目标节点集 T 的错误预测数量，如式(2)所示，其中 y_i 是目标主机入侵检测模型对节点 v_i 的分类结果（y_i= 1 表示恶意节点），$\mathrm{GC}(\cdot)$ 是基于 GNN 的节点分类器，$\mathrm{GC}(\cdot)_i$ 是节点 v_i 的分类结果（良性或恶意），T 是 $\boldsymbol{G}$ 中待攻击的目标节点集。在主机入侵检测场景中，T 只包含被目标主机入侵检测模型分类为恶意节点。

$$\boldsymbol{A'}=\mathrm{AK}(\boldsymbol{A})=\boldsymbol{A}+\delta \tag{1}$$

$$\max_{\boldsymbol{G'}}\left|\left\{\mathrm{GC}(\boldsymbol{A'},\boldsymbol{X'})_i\neq y_i, i\in T\right\}\right|$$
$$\text{s.t. } \mathrm{GC}(\boldsymbol{A},\boldsymbol{X})_i=y_i=1 \tag{2}$$

3.1.2　限制条件

与普通的针对 GNN 的对抗攻击不同，针对真实场景中的主机入侵检测模型的对抗攻击会有一些限制条件，本文主要考虑功能限制条件和成本限制条件 2 类。

功能限制条件：针对主机入侵检测模型的对抗攻击不能破坏恶意样本的恶意功能，这意味着在扰动溯源图时必须保持参与攻击的系统实体和系统事件不受影响。具体哪些系统实体和系统事件参与了攻击无法事先定义，因此只在输入溯源图中添加节点和边，而不删除节点和边。例如，如果删除图 1(a) 中的系统事件 (procdump64, 创建、读、写入, C:\Windows\Temp\lsass_dump-1.dmp)，则节点“procdump64”大概率会被分类为良性，但凭证转储功能也很可能会被破坏。

成本限制条件：攻击者需要在问题域中实现扰动后的溯源图特征张量，因此针对主机入侵检测模型的对抗攻击样本的实现成本很高。例如，要在输入溯源图中添加一个新的进程节点，攻击者需要实现读取文件、遍历注册表等大量的实际操作来在内核日志中具象化这个进程。为此，限制扰动后每个节点新增的一阶邻居节点不超过 b 个。

3.1.3　对抗攻击算法选型

本文选择如下 3 种黑盒对抗攻击算法来研究对抗迁移性。所有算法都只允许通过添加节点和边来实现结构攻击，且添加的数量需要满足限制条件。

随机采样攻击（RND）：给定一个目标节点 v，不断随机采样一个标签与 v 不同的节点 u，并将边 e_{uv} 添加到图结构中，直到目标节点 v 分类错误[22]。

图神经网络对抗攻击（Nettack, adversarial attack on neural network for graph data）：在 RND 的基础上进行了优化。首先，只生成与目标子图结构相似的扰动，确保扰动不易察觉。其次，设计了一个评分函数来评估逃逸概率[11]。

拓扑缺陷图注入攻击（TDGIA, topological de-

fective graph injection attack）：是一种基于图注入的攻击策略[23]。给定目标节点集T和要注入的新节点，使用拓扑缺陷选择策略选择T中与该新节点连接的目标节点v，然后进行注入。

3.2 对抗迁移性评估

在黑盒对抗攻击场景中，攻击者首先通过查询目标模型构建代理模型，其次针对代理模型生成对抗攻击样本，最后将对抗攻击样本迁移到目标模型上。因此，对抗迁移性是黑盒对抗攻击的重要前提。对抗迁移性可由多种因素造成。例如，对抗攻击样本通常位于模型的决策边界，而不同模型可能具有相似的决策边界[24]，此现象在针对GNN的对抗攻击中同样存在[25]。然而，此现象并不是绝对的。

$$\mathrm{Trans}\left(D_i \to D_j\right)=\frac{\left|\mathrm{mis}\left(D_i,D_j\right)\right|}{\mathrm{mis}\left(D_i\right)} \tag{3}$$

算法1 对抗攻击性评估

1) 初始化：溯源图样本集GS，一对主机入侵检测模型D_i和D_j，最大攻击预算Δ，$\mathrm{es}_i=0$，$\mathrm{es}_{ij}=0$
2) for 图集合GS中每一个溯源图gs_k do
3) $\mathrm{as}_k=\mathrm{gs}_k$ // 初始化对抗样本
4) for 当前图gs_k中每一个目标恶意节点v_{ki} do
5) $\mathrm{as}_k=\mathrm{perturb}(\mathrm{as}_k)$ // 应用扰动
6) while 判别模型D_i将v_{ki}判定为恶意 do
7) if $\mathrm{cost}(\mathrm{gs}_k-\mathrm{as}_k)<\Delta$ do
8) $\mathrm{as}_k=\mathrm{perturb}(\mathrm{as}_k)$
9) else
10) break
11) end if
12) end while
13) if 判别模型D_i将v_{ki}判定为良性 then
14) $\mathrm{es}_i=\mathrm{es}_i+1$
15) if判别模型D_j将v_{ki}判定为恶性then
16) $\mathrm{es}_{ij}=\mathrm{es}_{ij}+1$
17) end if
18) end if
19) end for
20) end for

为此，本文基于算法1所示的伪代码评估2.2节得到的8个主机入侵检测模型间的对抗迁移性。给定一对主机入侵检测模型D_i和D_j，首先基于某种对抗攻击算法为每个溯源图样本生成一个扰动版本。其中，perturb(·)是3.1.3节的3种对抗攻击算法（RND、Nettack、TDGIA）之一，而cost(·)则用于评估扰动限制。其次，计算2个主机入侵检测模型中被成功扰动的目标恶意节点的数量。最后，根据式(3)计算从D_i到D_j的对抗迁移性（记为$\mathrm{Trans}\left(D_i \to D_j\right)$），其中$\mathrm{mis}\left(D_i\right)$是被$D_i$错误分类的目标恶意节点集，$\mathrm{mis}\left(D_i,D_j\right)$是同时被$D_i$和$D_j$错误分类的目标恶意节点集。

3.3 多模型集成策略

如果Trans(A→B)值较低，说明模型A到模型B的对抗迁移性较低，这意味着能逃逸模型A的对抗攻击样本逃逸模型B的可能性较低。因此，如果能选择对抗迁移性较低的主机入侵检测模型，并设计合适的集成策略，则有望构建一个鲁棒的对抗攻击防御系统。给定一组主机入侵检测模型（记为EDS）和一个溯源图节点v_k，定义以下3种基础模型集成策略。

恶意投票策略：只要EDS中有一个模型将v_k识别为恶意，则v_k的最终分类结果为恶意。将该策略记为MV。

硬投票策略：v_k的最终分类结果为EDS中大多数模型的分类结果。该策略要求EDS中模型的数量为奇数。将该策略记为HV。

软投票策略：EDS中的每个模型都会输出将v_k分为恶意的概率（记为MP_k）和良性的概率（记为BP_k）。计算所有模型的平均MP_k和平均BP_k。如果平均MP_k>平均BP_k，则v_k的最终分类结果为恶意。反之，则为良性。将该策略记为SV。

基于上述基础集成策略，通过模型准备、模型选择、内部融合和外部融合4个步骤来构建多模型集成防御系统，如图3所示。

模型准备：根据3.2节计算得到的模型间对抗迁移性，采用贪心算法选出对抗迁移性最低的模型对。具体来说，给定得到的8个主机入侵检测模型$\mathrm{DS}=\left\{D_1,D_2,\cdots,D_8\right\}$，第一步，从主机入侵检测模型DS中选出$\mathrm{Trans}\left(D_i \to D_j\right)$和$\mathrm{Trans}\left(D_j \to D_i\right)$平均值最低的2个模型$D_i$和$D_j$。第二步，定义模型对

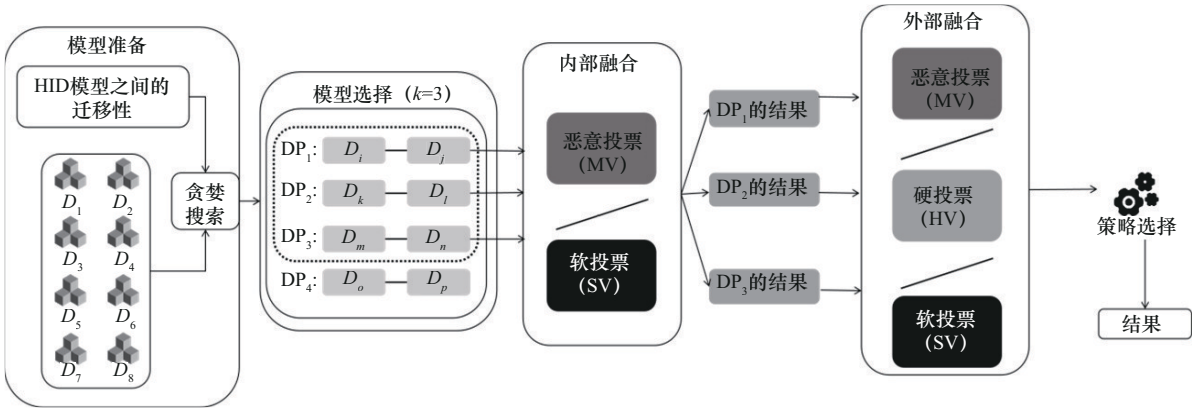


图3　多模型集成防御系统的构建流程

$DP_1=\left(D_i,D_j\right)$，将 DP_1 添加到模型对集合（记为 MPS）中，并从 DS 中删除 D_i 和 D_j。第三步，重复第一步和第二步，得到 4 个模型对（即 MPS = $\{DP_1, DP_2, DP_3, DP_4\}$）。

模型选择：假设需要选择 K 个主机入侵检测模型对（$k\leqslant 4$），则依次从 MPS 中选出。例如，如果 $k=2$，则选出 DP_1 和 DP_2。

内部融合：给定一个溯源图节点 v_k，对每个选出的模型对，采用基础集成策略 MV 或 SV 生成分类结果，则一共得到 k 个分类结果。

外部融合：采用基础集成策略 MV、HV 或 SV 对内部融合步骤得到的 k 个分类结果再进行集成，得到最终分类结果。

4　实验

4.1　实验数据集

为评测本文方法，需要一个包含大量具有节点级标注的溯源图样本的数据集。为采集符合需求的数据集，首先开发了一个面向 Windows 系统的内核日志数据采集工具（名为 KELLECT）。然后，在目标 Windows 主机上部署该工具，并采用由红色金丝雀公司（Red Canary）提供的自动安全测试工具 Atomic Red Team，在该主机上模拟各种 APT 攻击技术。在攻击模拟过程中，KELLECT 实时采集该主机自身产生的内核日志数据。Atomic Red Team 中包含大量模拟 APT 攻击的脚本，每个脚本对应一个 APT 攻击技术，每次模拟都会产生一个溯源图样本。

最终，数据集包含 473 个溯源图样本，涉及 9 种 APT 战术中的 26 种 APT 技术和 45 种 APT 子技术。溯源图样本的平均节点数和恶意节点数分别为 8 957 个和 4 个。该数据集已公开。

4.2　实验 1:主机入侵检测模型性能实验

本文实验采用 Ubuntu20.04 系统的服务器作为实验平台。服务器的 CPU 是 Intel(R)Xeon(R) Gold 5218 CPU, GPU 为 2 张 NVIDIA GeForce RTX 4090 显卡。具体硬件环境如表 2 所示。

表 2　硬件环境

名称	环境配置
处理器	Intel(R) Xeon(R) Gold 5218 CPU
显卡	NVIDIA GeForce RTX 4090×2
显存	24 GB×2
硬盘	8 TB
物理内存	125 GB

本文分别对 GCN、GraphSAGE、GAT、GIN 模型进行训练，循环迭代至验证集准确率收敛。训练过程中设置学习率为 0.001，使用 Adam 优化器；GCN 的层数为 3 层，每层随机失活率为 0.5；GraphSAGE 采用均值聚合器，邻居采样数分别为 25 和 10；GAT 的注意力头数设置为 8，合并方式为拼接；GIN 的 ε 可学习参数设为 0.2，多层感知机层数 3 层；所有模型的最大迭代轮次均为 200 轮，批处理大小为 16，损失函数采用交叉熵，正则化系数 λ=0.000 5。

本节实验测试 8 个主机入侵检测模型的性能，采用 3 折交叉验证来评估这些替代模型，并使用准确率、精确率和召回率作为评估指标。其中，精确率和召回率是针对恶意节点计算的。实验结果如表 3 所示。其中，f_1 和 f_2 分别代表结构视图特征

和行为视图特征，GCN_{f_1}表示基于结构视图特征和GCN作为底座GNN模型的主机入侵检测模型，GraphSAGE模型简称为SAGE。从实验结果可以看出，基于行为视图特征的模型的整体性能优于基于结构视图特征的模型，这可能是因为GNN自身具有学习结构特征的能力，而行为视图特征可以提供更多额外的语义信息；大多数情况下召回率高于精确率，说明基于溯源图学习的主机入侵检测模型可检测到绝大多数恶意节点，但有一定的误报率；主机入侵检测模型均表现出良好性能，证明了基于GNN的主机入侵检测方法的有效性。

表3 8种主机入侵检测模型的检测性能

模型	准确率	精确率	召回率
GCN_{f_1}	0.920	0.821	0.791
GCN_{f_2}	0.985	0.976	0.992
$SAGE_{f_1}$	0.912	0.889	0.893
$SAGE_{f_2}$	0.989	0.963	0.973
GAT_{f_1}	0.891	0.864	0.985
GAT_{f_2}	0.916	0.830	0.875
GIN_{f_1}	0.932	0.930	0.966
GIN_{f_2}	0.952	0.955	0.933

4.3 实验2：对抗迁移性测试实验

本节实验评测8个主机入侵检测模型间的对抗迁移性，评测流程如下。首先，对每个溯源图样本PG，采用3.1.3节的3种对抗攻击算法获得该样本的扰动版本PG′。其中，对抗攻击算法的目标为使得PG′最大化PG中被错误分类为良性的恶意节点数量。为全面评估对抗样本的多样性和攻击效果的统计稳定性，对于每个溯源图样本、每种对抗攻击算法、每个替代模型，均生成20个独立的扰动版本，通过预实验观察，当使用20个及以上的样本进行评估，目标入侵检测模型在对抗样本集上的召回率变化的均值与方差趋于稳定，继续增加样本量对评估结果稳定性的提升作用有限。不同对抗攻击算法对不同替代模型的攻击效果如表4~表6所示。其中，每个元素$A_k[D_i, D_j]$表示将针对替代模型D_i生成的扰动样本混入原始样本后，替代模型D_j的召回率变化，扰动样本是使用对抗攻击算法A_k生成的；向下的箭头表示与原始溯源图样本相比，召回率有所下降，例如，$RND[GCN_{f_1},GAT_{f_2}]=\downarrow 0.178$表示使用RND对抗攻击算法将针对$GCN_{f_1}$生成的扰动样本迁移到攻击$GAT_{f_2}$时，召回率下降了0.178。这里仅关注召回率，因为攻击者的目标始终是欺骗模型将恶意节点误分类为良性节点。

基于实验结果，可以得出以下结论。首先，矩阵对角线上的召回率下降幅度通常最大，这表明针对某一替代模型生成的扰动样本对其自身的攻击效果最好，但并非总是如此。这是因为黑盒扰动样本的生成并未参考特定替代模型的底层结构。其次，不同替代模型的鲁棒性存在差异。例如，GIN_{f_1}的鲁棒性最好，在对抗攻击下召回率平均下降了0.362；GCN_{f_2}是最容易被攻击的替代模型，召回率平均下降了0.538。再次，在黑盒对抗攻击算法中，

表4 基于RND的对抗攻击的效果

模型	GCN_{f_1}	GCN_{f_2}	$SAGE_{f_1}$	$SAGE_{f_2}$	GAT_{f_1}	GAT_{f_2}	GIN_{f_1}	GIN_{f_2}
GCN_{f_1}	↓**0.791**	↓0.644	↓0.539	↓0.310	↓0.403	↓0.178	↓0.469	↓0.015
GCN_{f_2}	↓0.189	↓**0.738**	↓0.395	↓0.661	↓0.096	↓0.396	↓0.047	↓0.578
$SAGE_{f_1}$	↓0.684	↓0.353	↓**0.733**	↓0.322	↓0.525	↓0.394	↓0.547	↓0.077
$SAGE_{f_2}$	↓0.391	↓**0.742**	↓0.248	↓0.654	↓0.203	↓0.608	↓0.086	↓0.533
GAT_{f_1}	↓0.672	↓0.579	↓0.706	↓0.429	↓**0.743**	↓0.206	↓0.578	↓0.095
GAT_{f_2}	↓0.240	↓0.718	↓0.272	↓0.642	↓0.199	↓**0.872**	↓0.078	↓0.574
GIN_{f_1}	↓**0.682**	↓0.481	↓0.642	↓0.155	↓0.610	↓0.475	↓0.672	↓0.082
GIN_{f_2}	↓0.136	↓0.594	↓0.179	↓0.618	↓0.078	↓0.446	↓0.046	↓**0.680**

表5　基于 Nettack 的对抗攻击的效果

模型	GCN_{f_1}	GCN_{f_2}	$SAGE_{f_1}$	$SAGE_{f_2}$	GAT_{f_1}	GAT_{f_2}	GIN_{f_1}	GIN_{f_2}
GCN_{f_1}	↓ **0.750**	↓ 0.660	↓ 0.688	↓ 0.416	↓ 0.602	↓ 0.525	↓ 0.647	↓ 0.302
GCN_{f_2}	↓ 0.512	↓ **0.962**	↓ 0.528	↓ 0.689	↓ 0.116	↓ 0.691	↓ 0.097	↓ 0.580
$SAGE_{f_1}$	↓ **0.682**	↓ 0.238	↓ 0.650	↓ 0.160	↓ 0.618	↓ 0.364	↓ 0.592	↓ 0.178
$SAGE_{f_2}$	↓ 0.200	↓ 0.524	↓ 0.058	↓ **0.591**	↓ 0.066	↓ 0.499	↓ 0.085	↓ 0.447
GAT_{f_1}	↓ 0.619	↓ 0.131	↓ 0.592	↓ 0.137	↓ **0.787**	↓ 0.298	↓ 0.564	↓ 0.071
GAT_{f_2}	↓ 0.211	↓ 0.486	↓ 0.196	↓ 0.505	↓ 0.057	↓ **0.603**	↓ 0.022	↓ 0.585
GIN_{f_1}	↓ 0.642	↓ 0.291	↓ 0.606	↓ 0.128	↓ 0.574	↓ 0.081	↓ **0.706**	↓ 0.089
GIN_{f_2}	↓ 0.182	↓ 0.501	↓ 0.188	↓ 0.518	↓ 0.058	↓ 0.614	↓ 0.605	↓ **0.744**

表6　基于 TDGIA 的对抗攻击的效果

模型	GCN_{f_1}	GCN_{f_2}	$SAGE_{f_1}$	$SAGE_{f_2}$	GAT_{f_1}	GAT_{f_2}	GIN_{f_1}	GIN_{f_2}
GCN_{f_1}	↓ **0.866**	↓ 0.582	↓ 0.650	↓ 0.488	↓ 0.809	↓ 0.108	↓ 0.509	↓ 0.353
GCN_{f_2}	↓ 0.328	↓ 0.694	↓ 0.325	↓ **0.698**	↓ 0.112	↓ 0.649	↓ 0.135	↓ 0.572
$SAGE_{f_1}$	↓ 0.660	↓ 0.251	↓ **0.757**	↓ 0.390	↓ 0.674	↓ 0.326	↓ 0.522	↓ 0.136
$SAGE_{f_2}$	↓ 0.271	↓ 0.757	↓ 0.453	↓ **0.813**	↓ 0.081	↓ 0.695	↓ 0.035	↓ 0.568
GAT_{f_1}	↓ 0.633	↓ 0.439	↓ 0.766	↓ 0.380	↓ **0.809**	↓ 0.101	↓ 0.603	↓ 0.011
GAT_{f_2}	↓ 0.107	↓ 0.701	↓ 0.291	↓ 0.684	↓ 0.034	↓ **0.792**	↓ 0.064	↓ 0.564
GIN_{f_1}	↓ 0.605	↓ 0.262	↓ 0.531	↓ 0.251	↓ **0.727**	↓ 0.611	↓ 0.709	↓ 0.362
GIN_{f_2}	↓ 0.296	↓ 0.585	↓ 0.366	↓ 0.593	↓ 0.231	↓ **0.878**	↓ 0.264	↓ 0.609

TDGIA 的效果最佳，平均降低了 0.471 的召回率。值得注意的是，与其他对抗攻击问题（如图像识别）的经验相比，恶意节点检测性能的下降幅度相对较小。这主要是因为，其他对抗攻击问题通常针对单一目标进行优化攻击，而本文需同时攻击溯源图中的所有恶意节点，寻求全局最优扰动。再次，在问题域中实现所有溯源图样本的扰动版本，并放入对抗攻击样本池（记为 ASP）。最后，采用 ASP 中的每个对抗攻击样本对每个主机入侵检测模型进行攻击，在此基础上基于式(3)计算从 D_i 到 D_j 的对抗迁移性 $\mathrm{Trans}\left(D_i \to D_j\right)$，其中 $\mathrm{mis}\left(D_i\right)$ 表示 ASP 中被 D_i 错误分类的恶意节点集。

评测结果如表 7 所示，其中每个元素 $A[D_i,D_j]$ 代表 $\mathrm{Trans}\left(D_i \to D_j\right)$。从实验结果可知，不同主机入侵检测模型间的对抗迁移性差异显著。例如，除相同的模型对之外，$SAGE_{f_1} \to GCN_{f_1}$ 的对抗迁移性最高，而 $GAT_{f_2} \to GIN_{f_1}$ 的对抗迁移性最低；具有相同特征视图的模型总是比具有不同特征视图的模型具有更高的对抗迁移性，说明不同的特征是导致对抗攻击样本无法迁移的最重要因素。

4.4　实验 3:对抗迁移性影响实验

本节实验评测对抗迁移性对模型检测性能的影响。具体来说，在对抗攻击样本池上测试所有主机入侵检测模型对的检测准确率、精确率和召回率，实验结果如图 4 所示。其中，编号较小的主机入侵检测模型对的对抗迁移性较低。由图 4 可知，随着对抗迁移性的升高（即编号增大），召回率总体呈下降趋势，这表明对抗迁移性越高的主机入侵检测模型对越容易受到对抗攻击，这验证了本文的关键假设。此外，SV 的下降趋势比 MV 更显著。这是因为有些情况下对抗迁移性的

表7　8个主机入侵检测模型之间的对抗迁移性

模型	GCN_{f_1}	GCN_{f_2}	$SAGE_{f_1}$	$SAGE_{f_2}$	GAT_{f_1}	GAT_{f_2}	GIN_{f_1}	GIN_{f_2}
GCN_{f_1}	—	0.812	0.626	0.402	0.508	0.171	0.504	0.096
GCN_{f_2}	0.490	—	0.509	0.914	0.265	0.851	0.216	0.510
$SAGE_{f_1}$	0.998	0.219	—	0.518	0.879	0. 689	0.619	0.207
$SAGE_{f_2}$	0.687	0.968	0.519	—	0.351	0.711	0.185	0.786
GAT_{f_1}	0.886	0.680	0.902	0.518	—	0.496	0.750	0.462
GAT_{f_2}	0.170	0.814	0.259	0.695	0.075	—	0.032	0.266
GIN_{f_1}	0.694	0.606	0.624	0.579	0.664	0. 619	—	0.573
GIN_{f_2}	0.517	0.566	0.450	0.598	0.477	0.848	0.540	—

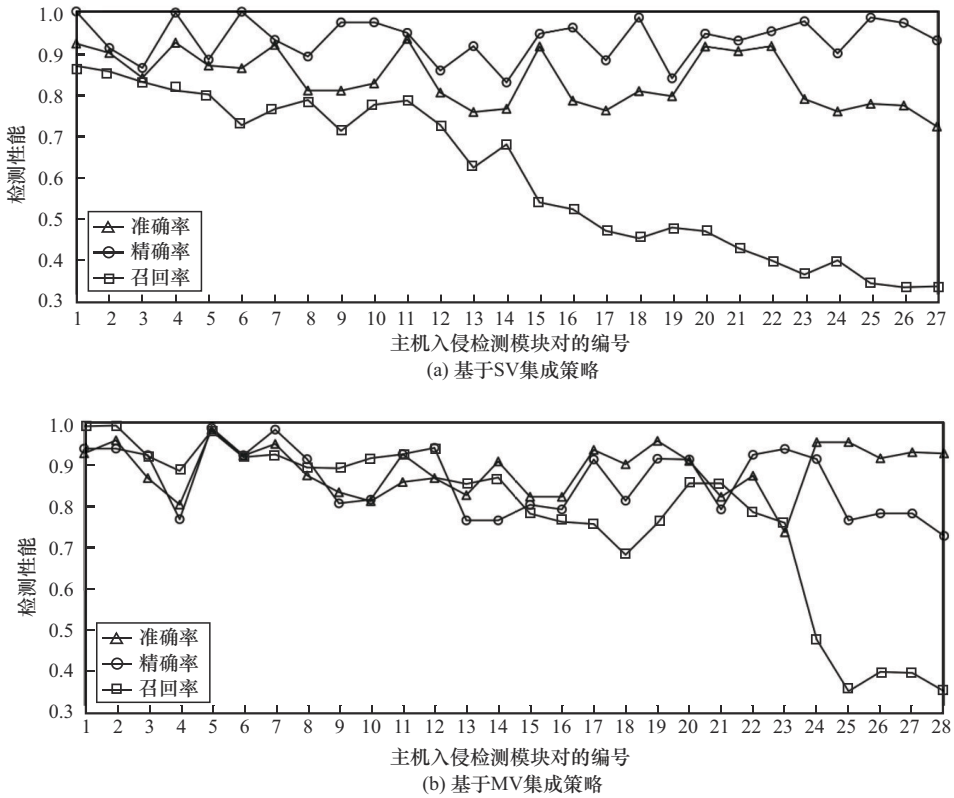


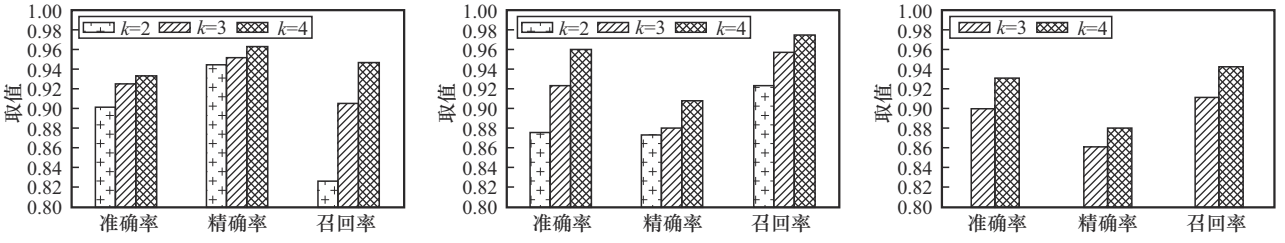
图4　不同主机入侵检测模型对的检测性能

升高尚不足以使得对抗攻击生效，但仍然会影响目标模型的分类结果概率分布。对抗迁移性对准确度和精确度（尤其是精确度）的影响不明显。这是因为本文的对抗攻击总是试图欺骗主机入侵检测模型将恶意节点误分类为良性节点，而非将良性节点误分类为恶意节点。

4.5　多模型集成策略实验

本节实验测试不同模型集成方式对对抗攻击防御效果的影响。在不同k值（即$k=2$、3、4）的前提下，通过组合不同的基础集成策略得到多种多模态集成方案，实验结果如图5所示。例如，MV(MV(DP_1) + SV(DP_2))代表将MV和SV作为DP_1和DP_2的内部集成策略，并将MV作为外部集成策略来集成DP_1和DP_2。

由图5可知，随着k值的增加，准确率和召回率都稳步上升，这说明集成更多的主机入侵检测模



(a) 基于SV外部集成策略　(b) 基于MV外部集成策略　(c) 基于HV外部集成策略

图5　多模型集成策略评估

型可增强对抗样本的防御能力。但当k值从2增加到3时，召回率有明显上升，而当k值进一步增加时，召回率的上升幅度明显降低。由于部署更多主机入侵检测模型会显著增加计算开销，因此选择$k=3$，以在防御能力和部署成本之间取得平衡。

4.6　对比实验

本节实验将本文方法与以下几个现有方法进行比较。

奇异值分解（SVD）：采用SVD获得原始溯源图样本的低秩近似版本，并将近似版本输入主机入侵检测模型。在原始溯源图上实施的扰动有较大概率不会反映在近似版本中，因此可防御对抗攻击[26]。

杰卡德（Jaccard）相似度：其依据是Jaccard相似度得分较低的2个节点之间的边更有可能是被恶意添加的，因此试图通过消除Jaccard相似度得分较低的边来消除潜在的对抗性扰动[27]。

图对抗训练（GraphAT, graph adversarial training）：指最广泛应用的对抗性训练策略[28]，其主要思想是在训练集中主动添加模拟对抗扰动样本，以增强对真实对抗攻击样本的鲁棒性。

图对抗防御框架ProGNN：利用正常图的固有特性（如低秩、稀疏、邻居节点特征相似）对输入图进行迭代重构，以获得干净的图[29]。

首先，本节实验中为每种防御方法专门生成对抗攻击样本，而非使用统一的对抗攻击样本池，这样的设置更符合真实情况。也就是说本文方法也被视为一个普通的主机入侵检测模型进行黑盒攻击。其次，采用Recall_{O}、Recall_{A}和FAR作为评估指标，分别代表原始样本（无对抗攻击样本）的召回率、对抗样本（无原始样本）的召回率和对抗攻击样本的误报率。需要说明的是，每个样本是一个溯源图，但主机入侵检测是在节点层面（即对每个溯源图样本中的每个节点进行检测），这意味着对抗攻击样本也包含良性节点。

推理时间成本对比和精确度实验结果对比分别如表8和表9所示，其中$k=3$。

表8　　推理时间成本对比

方法名称	推理时间成本/（ms·边$^{-1}$）
原始模型	0.057 4
SVD	0.408 6
Jaccard相似度	0.376 4
GraphAT	同原始模型
本文方法	0.367 4

首先，本文方法的Recall_{A}性能最好。由于所有现有方法都是基于单一模型设计的，这表明本文多模型集成策略在防御对抗攻击方面优于基于单一模型策略。其次，原始模型的Recall_{A}极低，这说明对抗攻击在没有防御机制的情况下对主机入侵检测模型的威胁极大。再次，所有方法的FAR没有明显差异，这说明防御机制对识别良性样本的影响不大。最后，与现有方法相比，本文方法在Recall_{O}方面也有优势。这表明在检测正常样本时，多模型集成策略比单一模型具有更强的泛化能力。然而，多模型集成策略的性能提升不可避免地伴随着更高的计算开销。如表8所示，在推理时间成本对比中，本文方法所需时间为0.367 4 ms/边。相较于无防御的原始模型，本文方法显著增加了推理延迟，其时间成本约为原始模型的6.4倍。这种开销主要源于需要串行运行多个基础GNN模型以构建集成防御系统。虽然现有防御方法如SVD和Jaccard相似度也存在时间成本增加，但本文方法的推理延迟增加是获取其卓越鲁棒性和泛化性能所需付出的主要代价。

表9 精确度实验结果对比

特征类型	方法	GraphSAGE			GAT			GCN			GIN		
		$Recall_O$	$Recall_A$	FAR	$Recall_O$	$Recall_A$	FAR	$Recall_O$	$Recall_A$	FAR	$Recall_O$	$Recall_A$	FAR
结构特征	原始模型	0.893	0.097	0.071	0.985	0.127	0.022	0.791	0.075	0.036	0.965	0.105	0.008
	SVD	0.824	0.239	0.188	0.981	0.284	0.022	0.695	0.285	0.071	0.936	0.354	0.012
	Jaccard相似度	0.928	0.378	0.063	0.954	0.248	0.039	0.879	0.238	0.088	0.971	0.411	0.073
	GraphAT	0.828	0.629	0.018	0.885	0.398	0.053	0.732	0.367	0.110	0.866	0.454	0.016
	ProGNN	0.944	0.623	0.013	0.942	0.772	0.048	0.637	0.411	0.034	0.719	0.577	0.097
行为特征	原始模型	0.924	0.012	0.072	0.972	0.023	0.012	0.875	0.074	0.036	0.932	0.085	0.091
	SVD	0.842	0.246	0.107	0.875	0.134	0.081	0.886	0.156	0.131	0.861	0.393	0.023
	Jaccard相似度	0.936	0.557	0.059	0.971	0.422	0.198	0.896	0.537	0.085	0.903	0.306	0.052
	GraphAT	0.858	0.772	0.073	0.797	0.517	0.121	0.882	0.711	0.028	0.924	0.557	0.078
	ProGNN	0.793	0.574	0.045	0.917	0.512	0.072	0.812	0.798	0.081	0.912	0.533	0.062
—	本文方法	0.956	0.885	0.037	0.992	0.816	0.079	0.916	0.812	0.085	0.973	0.829	0.089

5 结束语

本文尝试探索集成多个基于溯源图的主机入侵检测模型来防御对抗攻击，提出了一种面向主机入侵检测的多视图对抗攻击防御方法。通过研究和利用不同主机入侵检测模型间的对抗迁移性，为如何组合多个主机入侵检测模型提供了实践经验。在真实内核日志数据集上的实验表明，本文方法将检测对抗攻击样本的召回率提高了11.3%到53.4%。

参考文献：

[1] 徐志强, 文雨. 基于主机的高级持续威胁检测技术综述[J]. 计算机科学与应用, 2022(1): 233-251.
XU Z Q, WEN Y. A survey of host-based advanced persistent threat detection technology[J]. Computer Science and Application, 2022(1): 233-251.

[2] STOJANOVIĆ B, HOFER-SCHMITZ K, KLEB U. APT datasets and attack modeling for automated detection methods: a review[J]. Computers & Security, 2020, 92: 101734.

[3] 李元诚, 罗昊, 王欣煜, 等. 基于溯源图和注意力机制的APT攻击检测模型构建[J]. 通信学报, 2024, 45(3): 117-130.
LI Y C, LUO H, WANG X Y, et al. Construction of advanced persistent threat attack detection model based on provenance graph and attention mechanism[J]. Journal on Communications, 2024, 45(3): 117-130.

[4] JIA Z A, XIONG Y, NAN Y H, et al. MAGIC: detecting advanced persistent threats via masked graph representation learning[J]. arXiv Preprint, arXiv: 2310.09831, 2023.

[5] CHENG Z J, LV Q J, LIANG J Y, et al. Kairos: practical intrusion detection and investigation using whole-system provenance[C]//Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2024: 3533-3551.

[6] SUN L C, DOU Y T, YANG C, et al. Adversarial attack and defense on graph data: a survey[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 35(8): 7693-7711.

[7] ZÜGNER D, AKBARNEJAD A, GÜNNEMANN S. Adversarial attacks on neural networks for graph data[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM Press, 2018: 2847-2856.

[8] GOYAL A, HAN X Y, WANG G, et al. Sometimes, you aren't what you do: mimicry attacks against provenance graph host intrusion detection systems[C]//Proceedings of the 2023 Network and Distributed System Security Symposium. Piscataway: IEEE Press, 2023: 1-18.

[9] MUKHERJEE K, WIEDEMEIER J, WANG T H, et al. Evading provenance-based ML detectors with adversarial system actions[C]//Proceedings of the USENIX Security Symposium. Berkeley: USENIX Association, 2023: 1199-1216.

[10] WANG S, WANG Z L, ZHOU T, et al. THREATRACE: detecting and tracing host-based threats in node level through provenance graph learning[J]. IEEE Transactions on Information Forensics and Security, 2022, 17: 3972-3987.

[11] 陈晋音, 张敦杰, 黄国瀚, 等. 面向图神经网络的对抗攻击与防御综述[J]. 网络与信息安全学报, 2021, 7(3): 1-28.
CHEN J Y, ZHANG D J, HUANG G H, et al. Adversarial attack and defense on graph neural networks: a survey[J]. Chinese Journal of Network and Information Security, 2021, 7(3): 1-28.

[12] 武阳, 刘靖. 面向图像分析领域的黑盒对抗攻击技术综述[J]. 计算机学报, 2024, 47(5): 1138-1178.

WU Y, LIU J. A survey on black-box adversarial attack in image analysis[J]. Chinese Journal of Computers, 2024, 47(5): 1138-1178.

[13] PIERAZZI F, PENDLEBURY F, CORTELLAZZI J, et al. Intriguing properties of adversarial ML attacks in the problem space[C]//Proceedings of the 2020 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2020: 1332-1349.

[14] XIONG C L, ZHU T T, DONG W H, et al. Conan: a practical real-time APT detection system with high accuracy and efficiency[J]. IEEE Transactions on Dependable and Secure Computing, 2022, 19(1): 551-565.

[15] ZHU T T, YU J K, XIONG C L, et al. APTSHIELD: a stable, efficient and real-time APT detection system for linux hosts[J]. IEEE Transactions on Dependable and Secure Computing, 2023, 20(6): 5247-5264.

[16] AL-SADA B, SADIGHIAN A, OLIGERI G, et al. MITRE ATT&CK: state of the art and way forward[J]. ACM Computing Surveys, 2025, 57 (1): 1-37.

[17] 董程昱, 吕明琪, 陈铁明, 等. 基于异构溯源图学习的APT攻击检测方法[J]. 计算机科学, 2023, 50(4): 359-368.
DONG C Y, LYU M Q, CHEN T M, et al. Heterogeneous provenance graph learning model based APT detection[J]. Computer Science, 2023, 50(4): 359-368.

[18] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[J]. arXiv Preprint, arXiv: 1609.02907, 2016.

[19] HAMILTON W L. YING R, LESKOVEC J. Inductive representation learning on large graphs[C]//Proceedings of the 30th Annual Conference on Neural Information Processing Systems. Massachusetts: MIT Press, 2017:1025-1035.

[20] LIU Z Y, ZHOU J. Graph attention networks[C]// Proceedings of the 6th International Conference on Learning Representations. Vancouver: ICLR, 2018: 39-41.

[21] XU K Y, HU W H, LESKOVEC J, et al. How powerful are graph neural networks[C]//Proceedings of the 7th International Conference on Learning Representations. Vancouver: ICLR, 2019: 1-17.

[22] LI Y X, JIN W, XU H, et al. DeepRobust: a platform for adversarial attacks and defenses[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2021, 35(18): 16078-16080.

[23] ZOU X, ZHENG Q K, DONG Y X, et al. TDGIA: effective injection attacks on graph neural networks[C]//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. New York: ACM Press, 2021: 2461-2471.

[24] LIU Y P, CHEN X Y, LIU C, et al. Delving into transferable adversarial examples and black-box attacks[J]. arXiv Preprint, arXiv: 1611.02770, 2016

[25] MUJKANOVIC F, GEISLER S, GUNNEMANN S G, et al. Are defenses for graph neural networks robust[C]//Proceedings of the 35th Annual Conference on Neural Information Processing Systems. Massachusetts: MIT Press, 2022: 1-15.

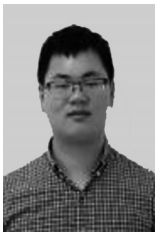
[26] ENTEZARI N, AL-SAYOURI S A, DARVISHZADEH A, et al. All you need is low (rank): defending against adversarial attacks on graphs [C]//Proceedings of the 13th International Conference on Web Search and Data Mining. New York: ACM Press, 2020: 169-177.

[27] WU H J, WANG C, TYSHETSKIY Y, et al. Adversarial examples for graph data: deep insights into attack and defense[C]//Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence. New York: ACM Press, 2019: 4816-4823.

[28] FENG F L, HE X N, TANG J, et al. Graph adversarial training: dynamically regularizing based on graph structure[J]. IEEE Transactions on Knowledge and Data Engineering, 2021, 33(6): 2493-2504.

[29] JIN W, MA Y, LIU X R, et al. Graph structure learning for robust graph neural networks[C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM Press, 2020: 66-74.

[作者简介]



王飞（1988– ），男，安徽宿松人，中国石油大学（华东）博士生，主要研究方向为人工智能、物联网、工业软件等。

钱可涵（2000– ），女，浙江绍兴人，浙江工业大学硕士生，主要研究方向为入侵检测技术、模型安全技术等。

吕明琪（1982– ），男，浙江杭州人，博士，浙江工业大学教授，主要研究方向为时空数据挖掘、数据驱动安全等。

朱添田（1992– ），男，浙江慈溪人，博士，浙江工业大学副教授，主要研究方向为网络攻击检测等。

陈鸿龙（1984– ），男，福建南安人，博士，中国石油大学（华东）教授，主要研究方向为智能物联网、人工智能安全、安全与隐私保护等。