SubstratumGraphEnv：用于建模系统攻击路径的强化学习环境（RLE）

Bahirah Adewunmi[1], **Edward Raff，博士**[2], **Sanjay Purushotham，博士**[1]
[1] 马里兰大学巴尔的摩县分校 [2] CrowdStrike b280@umbc.edu,
edward.raff@crowdstrike.com，psanjay@umbc.edu

arXiv:2603.01340v1 [cs.CR] 2 Mar 2026

摘要

自动化网络安全分析，尤其是识别潜在攻击路径，面临着重大挑战。这在一定程度上是因为系统事件具有顺序性、相互关联性和演变性，而大多数人工智能（AI）技术难以对其进行有效建模。本文提出了一种强化学习（RL）环境生成框架，用于模拟 Windows操作系统上执行的进程序列，从而能够动态建模系统中的恶意进程。该方法使用图表示对操作系统状态及其转换进行建模。此图由开源系统监视器（Sysmon）日志生成。为应对系统事件类型、字段和日志格式的多样性，我们开发了一种机制，用于从 Sysmon 日志中捕获并建模父子进程。我们构建了一个 Gymnasium 环境（*Sub-stratumGraphEnv*），以建立强化学习环境的可感知基础；同时还构建了一个定制的 PyTorch 接口（*SubstratumBridge*），用于将 Gym-nasium 图转换为深度强化学习（DRL）观测和离散动作。图卷积网络（GCN）具体表示图的局部和全局状态，并将其输入优势演员-评论家（A2C）模型中不同的策略头和评论家头。这项工作的核心贡献在于设计了一种新颖的深度图强化学习环境，可自动将顺序发生的用户和系统事件转换为模型输入，为网络安全分析提供关键上下文。本研究为未来探索训练参数调整和高级奖励塑造奠定了基础，同时也揭示了哪些系统事件属性对训练自主强化学习智能体至关重要。

引言

绝大多数网络安全事件都源于人为因素：88% 的安全漏洞与人为错误有关（Hancock 2022），80% 涉及凭据遭到泄露（Hylender等，2025）。大量此类攻击利用了错误配置；仅云配置错误就导致了 23% 的安全事件，这一点便是明证（SentinelOne2025）。自动化网络安全分析，尤其是绘制潜在攻击路径，也颇具挑战，因为系统事件具有复杂的顺序特性，而相互关联的系统事件序列往往异常冗长，这进一步加大了难度。传统的 AI 建模也面临困难，原因不只是数据具有序列性，还在于关键攻击序列的属性维度极高，且关联比常见的基于序列的 AI 模型所能有效处理的情况更为复杂。传统序列模型，如循环神经网络（RNN）及其变体（例如长短期记忆网络、门控循环单元），虽然能够处理时间序列数据，但往往无法捕捉系统进程日志中固有的复杂、非线性且通常呈图结构的依赖关系。

我们需要更复杂的方法，例如能够从根本上对关系数据建模的图神经网络（GNNs）、借助强大的注意力机制来捕捉长程依赖和丰富上下文嵌入的 Transformer，或通过与这些复杂动态环境交互并探索环境来学习最优策略的深度强化学习（DRL）。

尽管强化学习和基于图的方法已在网络安全领域得到各种应用，但我们的工作独特之处在于，直接将 Sysmon 日志中的原始高维操作系统事件数据建模为用于深度强化学习的动态基于张量的图。我们的主要贡献是开发了一种新颖的深度图形强化学习环境，可将 Sysmon 日志中的原始高维操作系统事件数据转换为基于张量的图，以便训练基于神经网络的自主智能体。这使强化学习智能体能够理解并学习具体的系统进程路径，这对于客观审计和评估系统安全态势至关重要。

本研究获得的见解将有助于设计更复杂的观测状态，并设计和训练强化学习算法，以理解恶意和良性行为所导致的细微却至关重要的时间变化，从而支持未来更先进的分析和检测工作。

文献综述

尽管强化学习和基于图的方法在网络安全领域有多种应用，但我们的工作独特地解决了直接对高维进程级事件数据进行建模这一挑战。我们的文献综述发现了一个显著的空白：

版权所有 © 2026，人工智能促进协会（www.aaai.org）。保留所有权利。

目前没有可直接比较的方法能够在应用或进程层面识别攻击路径、从观测到的进程依赖关系中抽象出攻击路径，或生成数据结构，使自主智能体能够学习并遍历系统进程路径。由于企业主机事件日志十分稀缺——这通常是隐私和安全问题所致——所有相关研究都不得不采用基于网络流或人工整理的专业知识（例如，入侵检测系统[IDS] 规则和已知漏洞）的强化学习动作空间。

现有将低级系统事件映射到强化学习状态、动作和奖励的方法差异很大，且往往缺乏标准化方案。许多方法会抽象掉底层系统进程。例如，Maeda 和 Mimura（2020）使用一组系统参数而非细粒度进程来建模后渗透战术，而 Harries 等人（2020）则采用符号树来建模用户界面交互。这些研究针对的是不同形式的系统活动，与我们建模的原始事件序列有所区别。

在攻击检测领域，一些研究也应用了强化学习，但抽象层次有所不同。Deokar 和 Hazarnis（2012）使用强化学习关联来自多个来源的日志文件，以确认攻击；Xu 和 Xie（2005）则通过将系统调用序列建模为马尔可夫奖励过程，利用强化学习进行入侵检测。这些方法虽有价值，却未涉及智能体在动态的、基于图的环境中采取明确的探索性动作。同样，在渗透测试领域，Yang 等人（2022）和 Elderman 等人（2017）开发了用于发现攻击路径的强化学习框架，但其状态表示通常是网络和主机信息矩阵，而非从原始系统日志中提取的、关于进程和事件的详细图结构关系。

近年来，研究开始将深度强化学习与图表示结合，用于网络安全任务，但这类研究通常处于较高的抽象层次。Gueriani 等人（2024）和 Jamshidi 等人（2025）对物联网中基于深度强化学习的入侵检测系统进行了综述，重点介绍了深度强化学习，有时也涉及图神经网络的应用，但往往依赖网络流特征或聚合数据，而非细粒度的原始操作系统（OS）事件。同样，Cody 等人（2022）和 Li 等人（2024）的研究利用强化学习识别攻击路径，但其图表示描述的是网络拓扑和漏洞，而非本研究关注的核心内容——动态进程关系。尽管 Li 等人（2024）采用了图卷积网络，但其研究聚焦于特定信息物理系统中的数据完整性，并未探讨如何将通用操作系统事件流转换为基于张量的图，以学习系统行为。Wang 等人（2025）也推动了强化学习在网络防御中的应用，但并未深入研究如何根据原始日志对基于主机的系统进程进行细粒度建模。

正如 Zheng 等人（2021）所指出的，这些强化学习应用面临的一项关键挑战是设计有效的奖励函数。在复杂的序列环境中训练智能体时，会遇到奖励稀疏和延迟的问题，而在大型状态空间中，这一问题会更加严重。此外，尽管 Kujanp¨a¨a 等人（2021）研究了权限提升，但将这类高层安全行为映射到 Sysmon 等日志中的底层事件仍然十分复杂。仿真环境的开发对于将强化学习应用于安全领域至关重要，但通用的基于图的训练框架通常无法捕捉真实操作系统事件数据所固有的特定语义和序列依赖关系（Ergun 等人，2023；Zhao 等人，2020；Liu 等人，2025）。

方法

数据集

可靠配置的高保真企业主机事件日志仍极为稀缺。Kenyon 等人（2020）发现，中心注册表不一致、威胁不断演变导致数据迅速过时，以及旧数据集中大量匿名化等问题进一步加剧了这一状况。由于 Windows 和 Linux 系统默认不会启用此类日志配置，父子进程操作系统事件数据集也更加难以获取。

为体现这些数据方面的限制，本研究利用了两个由网络攻击仿真生成的开源 Sysmon 数据源。对于 Windows 操作系统，Sysmon 擅长捕获详细的父子进程关系。第一个数据源名为 BRAWL，Kemmerer 和 Wampler（2018）将其描述为一个云托管企业网络，其中 MITRE 的 Caldera 充当攻击方。BRAWL 中的事件可直接映射至 MITRE 网络分析存储库（CAR），从而能够捕获凭证转储和横向移动等各种攻击后行为。为使方法论稳健性的评估更加多样化，尤其是在构建网络和 Gymnasium 图方面，研究还纳入了 Cerberus Traces 数据集（Pratomo，2023）。该数据集提供了更多 Sysmon 时序事件，这些事件采集自真实的 Windows 环境，涵盖恶意和良性软件的执行。

图设计

我们的方法将系统活动建模为从处理后的 Sysmon 日志中提取的动态有向图。这种设计的根本目的有二：首先，准确且结构化地捕捉进程因果依赖关系（例如父子关系），这些关系在原生操作系统日志中往往被掩盖或缺失，从而揭示由系统配置错误或被忽视的安全参数所导致的复杂攻击路径。其次，图结构提供了形式化定义的状态表示，与马尔可夫决策过程（MDP）框架相契合。这种转换将原始日志数据转化为结构化且可遍历的格式，这对于构建有效的深度强化学习（DRL）环境和奖励函数至关重要，使自主智能体能够学习识别并遍历构成对抗事件的序列。对生成图的分析表明，后续的奖励函数必须优先鼓励长距离遍历，并激励智能体在分散的节点集合中发现关键的序列依赖关系。

1. **节点**：表示离散的系统状态或实体（例如，通过可执行文件路径和标识符（ID）识别的运行中的进程、已执行命令）。

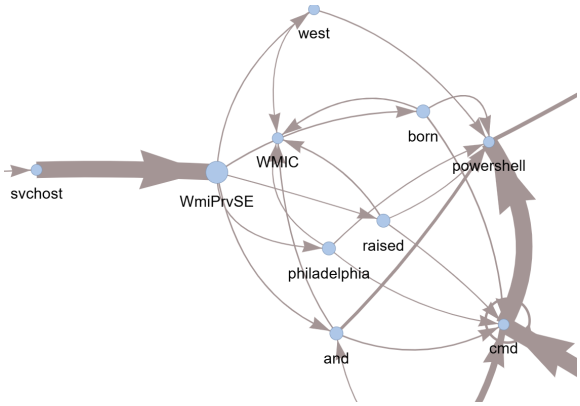


图 1：这张 BRAWL 数据集的局部网络快照揭示了系统进程图稀疏且不具同配性的特征。节点标签涵盖了*svchost* 等常见 Windows 进程，以及取自《新鲜王子妙事多》主题曲的字符串；这些都是直接从数据集的 Sysmon 日志中提取的原始标识符。尽管 *powershell* 等常见进程有许多连接（粗边），但较低的聚类系数表明，强化学习智能体不能依赖稠密的局部模式，而必须学会在这些反直觉的稀疏路径中导航，以识别完整的攻击链。

2. **边**：表示状态之间的转换，体现因果联系，例如父进程创建子进程，或进程执行命令。 3. **属性**：边包含交互频率（用作权重）和上下文元数据等属性。节点包含进程名称、父进程 ID、完整性级别、事件持续时间、进程 ID、映像路径、主机名、文件路径等属性。

父子关系是以 Sysmon 事件 ID 1（进程创建）作为父进程建立的。子节点则取自 Sysmon 事件 ID，例如事件 ID 2（文件创建时间已更改）、5（进程终止）、11（文件创建）、12（注册表对象创建/删除）和 13（注册表值设置）。对 BRAWL 和 Cerberus Traces 数据集进行处理后，提取出这些事件作为节点。

图 1 展示了 Sysmon 日志中父进程与子进程之间的有向关系，其中源节点是父进程，终端节点是子进程。从视觉上观察到的低聚类程度表明，仅依赖密集局部事件模式的奖励函数效果不佳。相反，强化学习状态表示应聚焦于独立事件及其紧邻的顺序关系，而智能体必须学会遍历更长、更稀疏的路径，以实现安全相关目标。

收集并筛选后的事件数据被转换为 NetworkX 有向图，以 有向图 对象表示。此过程将日志中识别出的关系解释为表示系统实体的节点之间的有向边。节点和边的属性从日志字段中提取，并附加到相应的图元素上。

这些 NetworkX 图为系统事件序列提供了结构化表示。

作为自定义 Gymnasium 环境开发的，*Sub-stratumGraphEnv* 保留了图的连通性信息，并将状态空间建模为有向图。它将内部 NetworkX 图状态转换为 *gymnasium.spaces.Graph* 字典，从而为图表示与深度强化学习框架搭建接口，并为节点和边属性提供标准化的字典布局。每个条目存储图快照的一个组成部分：节点特征矩阵、边索引对，以及节点数和边数的标量计数。

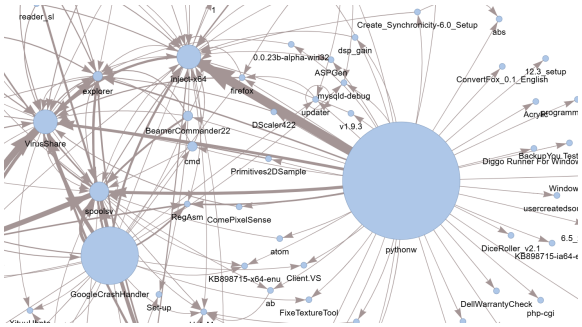


图 2：Cerberus Traces 的这一快照显示，进程图密集且复杂，少数进程（如*pythonw*）充当活动的中心枢纽。尽管连通性很高，但网络的低聚类系数表明，关键攻击序列并不局限于局部密集的事件模式。因此，强化学习环境必须引导智能体在庞大、分布式的状态空间中探索，以发现重要的序列连接。

图分析

我们对 BRAWL 和 Cerberus Traces 数据集中的 Sysmon 事件图进行了结构分析。我们观察到，这些图具有高度稀疏性、较低的聚类系数以及不相称的混合模式（详见表 1 和表 2），这表明利用进程依赖关系的攻击序列不会形成密集的局部簇。这些网络特征体现了攻击链过程：网络环境具有稀疏且延迟的奖励挑战。就自主智能体行为而言，攻击者通常需要在较长时间内执行大量连续动作才能实现目标。因此，这项分析表明，需要设计精细的奖励函数，引导智能体在庞大的状态空间中学习并遍历漫长、分散且关键的路径。

移除无子节点的父节点后，从 BRAWL 数据集导出的图包含 104 个节点和 227 条边，密度为 2.12×10^{-2}。Cerberus Traces 数据集生成了一个更大的图，包含 6,033 个节点和 7,446 条边，密度为 2.05×10^{-4}。对每个图的聚类系数进行分析后发现，两种网络的聚类系数总体偏低。聚类系数衡量局部互连性（即一个节点的邻居彼此相连的可能性），这一分析也凸显了稠密且

事件的局部聚类。BRAWL 网络总体上显示出较低的平均最近邻度，尤其是对于高度节点，这表明存在异配混合模式，即频繁发生的事件连接到较少发生的事件。Cerberus Traces 网络呈现出

Cerberus Traces		BRAWL	
Key 度 (k)	平均最近邻居度（k_{nn}）	Key 度（k）	平均值最近邻居度 (k_{nn})
1	487.94	1	0.10
2	103.11	2	0.77
3	152.66	3	0.46
4	55.98	4	0.67
15	63.08	5	0.34
30	0.36	9	0.45
55	2.73	15	0.18
97	2.40	18	0.06
109	2.69	20	0.30
152	13.22	24	0.01
300	0.83	56	4.35
23	4.55	62	0.01
24	7.41	23-24	0.01 (k=24)
最大值 (2566)	0.58		—

表1：平均最近邻度差异很大且通常偏低，并且存在异配模式——即高阶节点连接到低阶邻居——这些现象表明，奖励不能仅依赖于事件的局部稠密簇

更复杂的模式。尽管一些高度节点呈现负向关联，许多低度节点却连接到度极高的邻居。这表明，某些外围事件可能与核心系统活动紧密相关。这些连通模式凸显了强化学习智能体必须学会应对的多样结构情境。表1比较了 Cerberus Traces 和 BRAWL 数据集中的节点度及平均最近邻度，结果表明，奖励函数必须经过精心设计，才能在连通性差异极大的节点间传播信号。策略需要借助奖励，有效引导智能体遍历漫长、稀疏且不符合直觉的路径，以实现安全相关目标。

入度中心性分析识别出接收大量入边的节点（代表交互关系中的依赖方或目标）。表2列出了两个数据集中入度中心性最高的节点。BRAWL 网络中，*conhost*、*cmd* 和 *power-shell* 等核心系统工具的入度中心性相对较高，这表明这些进程周围的交互模式更加集中。

相比之下，规模更大的 Cerberus Traces 网络中，排名靠前的节点入度中心性得分显著较低，这表明其交互模式更加分散，且缺乏高度集中的共同入口点。这直接影响强化学习环境的设计，因为稀疏、低中心性的图结构使简单奖励机制的设计更加复杂。仅基于

进程/节点名称	入度中心性值	数据集
conhost	0.13600	BRAWL
Cmd	0.06800	BRAWL
PowerShell	0.05800	BRAWL
Reg	0.05800	BRAWL
Net	0.04900	BRAWL
WMIC	0.04900	BRAWL
Google 更新	0.02900	BRAWL
ProxyEnable	0.00862	Cerberus Traces
inject-x64	0.00381	Cerberus Traces
is32bit	0.00381	Cerberus Traces

表2：与Cerberus Traces相比，BRAWL中显著更高的中心性得分（例如*conhost*、*cmd*）揭示了两种截然不同的交互模式：BRAWL中为集中式，Cerberus中为分布式。这种结构差异证实，需要一种能够针对分散的关键节点、而不仅仅是单个高度连接目标的奖励函数。

抵达单个高度连接节点来设计奖励机制将收效甚微。相反，有效的奖励结构必须能够在智能体穿越数量更多、分布更分散的关键节点时给予奖励，或奖励其在多个连接度较低的节点上取得的累积效果，以反映 Cerberus 数据集中交互更加分散的特性。

总体而言，这些图属性（稀疏性、多样的度连通模式、低聚类）为强化学习环境的设计提供了依据，凸显了以下必要性：1. 能够处理稀疏、可能为高维输入的状态表示；2. 能够捕捉分布式模式并引导对序列性、非局部聚类事件进行学习的奖励函数；以及3. 适用于在局部连通性有限且状态/动作空间较大的环境中进行探索的强化学习算法。

强化学习环境设计

为规避Gymnasium库与*torchrl.envs.GymEnv*的不兼容问题，我们开发了*SubstratumBridge*。在撰写本文时，PyTorch的*gymenv* 包装器与基于Gymnasium 1.0或更新版本构建的图不兼容，原因是行为不可预测。/ emphSubstratumBridge将图属性转换为PyTorch张量格式，从而能够基于系统进程属性（如进程事件完整性级别、事件持续时间和用户信息）进行神经网络训练。这种对细粒度系统事件数据的关注，优先学习与攻击检测和攻击路径特征相一致的安全目标，同时兼顾图内的序列依赖关系。这改进了先前工作中较为抽象的方法，那些方法往往抽象掉底层系统进程，或侧重于用户界面交互。

马尔可夫决策过程（MDP）为建模网络攻击路径提供了强大的框架：它将运行中的进程视为离散的时间点，并将这些状态之间的转换视为由逻辑关联的子事件驱动。父子事件关系是此类因果联系的典型示例，其中包含了预测和理解系统内事件流动所需的关键信息。

MDP 通常定义为一个元组（S,A,P,R,γ）：

$$V^*(s) = \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} \mathcal{P}(s'|s,a) \left[\mathcal{R}(s,a,s') + \gamma V^*(s') \right] \quad (1)$$

在我们的实验设置中，任意时刻的状态（$s \epsilon S$）由 Sysmon 事件图中的当前节点（进程）定义。动作（$a \epsilon A$）是对目标节点的离散选择，并通过节点邻接矩阵进行有效性检查。V^*（s'）是从状态 s'开始能够获得的最大奖励期望累积未来奖励。

奖励函数（$R(s,\ a,\ s')$）旨在引导智能体在稀疏复杂图环境中探索。基线奖励算法通过逐步惩罚、对无效且非相邻的移动施加严厉惩罚，以及针对完整性级别进展和入度中心性设置递增奖励，提供了密集的学习信号。

算法1：基线奖励函数 $\mathcal{R}(s,\ a,\ s')$

输入： 当前状态 s，动作 a（目标-get Node s'），图 $\mathcal{G}$，常数 $\alpha, \beta, R_{step}, R_{term}, R_{escalate}, R_{downgrade}$
输出： 奖励 R
$R \leftarrow R_{step}$ 效率的固定负步数惩罚
效率
如果状态 s 无效 **或**目标 s' 越界 **则**
    $R \leftarrow -1.0$
    返回 R
结束 if
如果 边 $(s,\ s')$ 在 $\mathcal{G}$ **中不存在，则**
    $R \leftarrow R_{step} - 0.5$
    返回 R
否则
    $w \leftarrow$ 边权重$(s,\ s')$
    $R \leftarrow R + w$
    $R \leftarrow R -(\alpha \cdot w)$
    $I_s \leftarrow$ 完整性级别(s)
    $I_{s'} \leftarrow$ 完整性级别(s')
    如果 $I_{s'} > I_s$ **则**
        $R \leftarrow R + R_{escalate}$
    否则如果 $I_{s'} < I_s$ **则**
        $R \leftarrow R + R_{downgrade}$
    结束如果
    $C_{s'} \leftarrow$ 入度中心性(s')
    $R \leftarrow R +(\beta \cdot C_{s'})$
    如果 s' 是终端节点 s_{term} **则**
        $R \leftarrow R + R_{term}$
        回合终止 $\leftarrow$ True
    结束如果
结束如果
返回 R

算法1概述了奖励函数。该函数旨在直接应对在 Cerberus Traces 和 BRAWL 图分析中观察到的稀疏性、低聚类程度以及多样的度连通性。恒定的负步惩罚可激励智能体在漫长、分散的路径中提高效率；同时，严厉的惩罚 -1.0

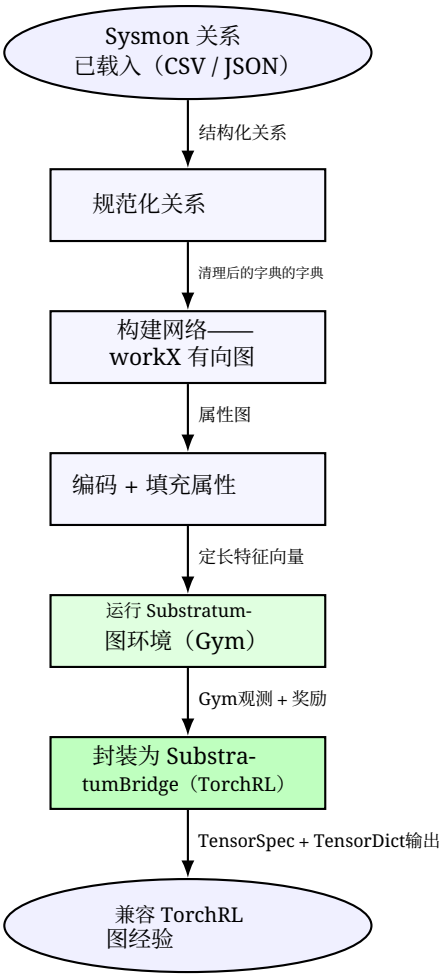


图3：从原始 Sysmon 关系到可供 TorchRL 使用的图观测的高级流程。

以及 -0.5 可遏制错误动作，并强制遵守图的严格邻接规则。为克服良性噪声普遍存在的问题，并获取有助于检测攻击路径的重要属性，边遍历惩罚会降低智能体频繁选择相同边的倾向，而完整性级别奖励（+100提升）和目标节点中心性奖励则专门引导智能体关注重要的网络进程。将这些属性加入奖励算法，是为了弥补分散的交互模式所带来的影响，因为这类模式会使稀疏图中的学习更加困难。最后，在实验开始时到达终端节点集可获得大额 $+5000$ 奖励，从而强化这一行为，提供非局部目标，并设定奖励信号以应对环境稀疏性。

后续调整将无效移动的惩罚提高了 10 倍，以强烈抑制不理想的动作。同时，将移动到较低完整性级别的惩罚大幅降低了 100 倍，以赋予智能体更大的灵活性。最值得注意的是，边权重频率惩罚被基于同一节点选择频率的惩罚所取代；每当在一次.....中重新访问某个节点时，该惩罚就会逐步增加

回合。

基于图的演员-评论家模型设计

为满足处理图网络中固有的稀疏、非局部依赖关系的需求，我们选择了基于图的优势演员-评论家（A2C）架构设计。图卷积网络层至关重要，因为它们通过处理每个节点的特征，并在整个图的节点邻接矩阵中迭代聚合信息，捕获局部特征和全局图拓扑结构。通过全局池化，图卷积网络层的输出被整合为一个固定大小的向量，用以表示整体图状态。这一过程将动态的图状态转换为张量格式，验证了所设计的深度强化学习环境和观测空间的可行性，可用于借助基础强化学习智能体开展客观的流程级安全分析。

1.**初始节点嵌入**：表示离散系统状态或实体（例如，通过可执行文件路径和标识符（ID）识别的运行中的进程、已执行命令）。

2. **图卷积层**：两个 PyTorch GCNConv 层，大小均为 m。通过聚合来自邻居的信息来处理节点嵌入，以捕获局部及更广范围的图拓扑上下文。这些层之后均应用线性整流单元（ReLU）激活函数，并加入随机失活以进行正则化。

3. **图汇总**：全局均值池化层将图卷积网络学习到的节点表示聚合为一个大小为 m 的向量，用于表示整个图状态。

4. **演员**：线性层。将汇总后的图表示映射为一个向量，其中包含用于选择离散动作空间中每个可能动作（n 个可能动作）的未归一化分数或对数几率，从而定义智能体策略。

5.**评论家**：线性层。将整合后的 m 大小的图表示映射为一个标量值，即模型对从当前状态可获得的期望累积未来奖励的估计。

GCN 的训练过程使用 Adam优化器来最小化组合损失函数。Adam 是一种流行且稳健的选择，适用于训练多种深度神经网络，因为它结合了均方根传播（RMSprop）和动量的优点。损失函数包含三个主要部分：

1. **策略损失**（L_p）：Mnih等人（2016年）提出了策略损失的算法，其设计旨在提高能够带来正优势的动作的概率，并降低会导致负优势的动作的概率。对于包含 N 个样本（状态-动作对）的批次，策略损失通常计算如下：

$$L_p = -\frac{1}{N}\sum_{i=1}^{N}\log \pi(a_i|s_i)A(s_i,a_i) \tag{2}$$

其中，$\pi(ai|s_i)$ 表示根据演员策略在状态s_i 下采取动作 a_i 的概率，而 $A(s_i,a_i)$ 表示在状态 s_i下采取动作 a_i 的估计优势。

2. **价值损失**（L_v）：价值损失使用 PyTorch（Barnes 2021）的均方误差（MSE）损失实现。它衡量评论家的估计值与环境中实际观测回报之间的差异。价值损失的强度由价值损失系数调节。对于一批 N 个状态，价值损失（L_v）为：

$$L_v = \frac{1}{N}\sum_{i=1}^{N}(V(s_i) - R_i)^2 \tag{3}$$

其中，$V(s_i)$ 是评论家对状态 s_i 估计的值，而 R_i 是从状态 s_i 开始计算得到的目标回报（折扣累积奖励）。

3. **熵正则化**（L_e）：PyTorch 的熵项旨在通过提高智能体动作选择的多样性来促进探索（Fritz 2017）。这有助于防止策略过快收敛、变得过于确定，或形成次优策略。正则化的强度由一个系数控制。

$$L_e = -\sum_{a\in A}\pi(a|s_i)\log(\pi(a|s_i)) \tag{4}$$

其中，$\pi(a|s_i)$ 是根据策略 π 在状态 s_i 下采取动作 a 的概率，A 是所有可能动作的集合，β 是熵系数，用于控制正则化的强度。

优化器最小化的总损失是这些组成部分的加权和：

$$L_{Total} = L_p + c_v L_v + c_e L_e \tag{5}$$

实验设计

此实验设计的根本目的是测试并衡量深度强化学习环境及其观测空间的可行性和训练稳定性。为了选取一条通往目标进程的合理攻击路径，我们通过程序识别了每个网络中的最长路径。由于这两个图中都存在有向环，路径选择过程使用 NetworkX 的 *dag-longest-path* 函数，在每个网络的凝缩图上进行计算。对于 BRAWL 数据集，这确定了一条从 *TrustedInstaller* 进程到某个 IPv6 地址的路径。在 Cerberus Traces 数据集中，则识别出一条始于 *execsc*进程并终于*Procmon*的路径。

实验运行环境为 Debian 12 操作系统，配备 2 TB 磁盘容量、64 GB 随机存取存储器（RAM）、一块配有 24 GB 显存的 NVIDIA Geoforce RTX 3090 图形处理器，以及一颗拥有 16 个核心和 32 个逻辑处理器的 AMD Ryzen 9 7950X 中央处理器（CPU）。

实验旨在评估强化学习智能体在图形环境中的训练稳定性，并探索各种模型超参数对强化学习环境性能的影响。这使实验在关键超参数的预定义数值集合上运行。具体而言，实验针对这些超参数值的每种唯一组合进行初始化：

1. 模型输出维度：图卷积网络输出的维度。2. 价值损失系数（c_v）：控制整体损失函数中价值损失分量的权重，可选值为 0.5、1.0 和 2.0。3.熵系数（c_e）：通过对熵正则化项加权来影响探索与利用的权衡，使用的值为 0.001、0.01、0.05 或 0.1。4.学习率：决定训练期间模型权重的更新步长，探索的速率为 0.0001、0.001、0.005 和 0.01。5. 总训练步数：定义每次运行的总训练步数（1000、2000 或 5000）。6. 节点频率惩罚尺度：该参数取值为 0.5、5.0 和 20.0，用于调整在一个回合内重复访问节点所施加的惩罚，以促进对网络的探索。7. 并行环境数量选项：实验运行中使用的并行环境数量在 2、3、12、16 和 32 之间变化

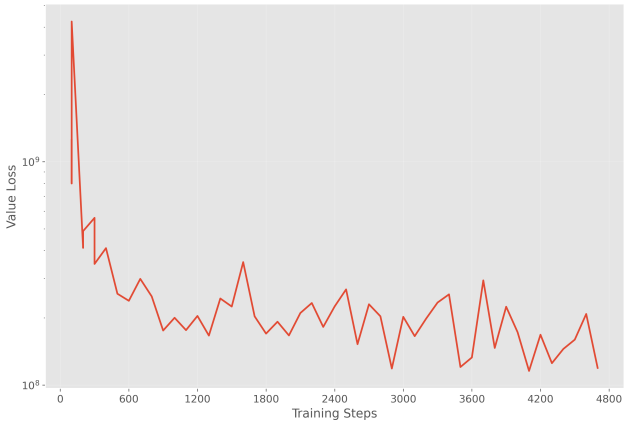


图 4：Cerberus 深度强化学习环境上价值损失与训练步数的关系。价值损失下降表明评论家正在学习准确预测价值

折扣因子（γ =0.99）和每次更新前的步数（*trainer-n-steps = 3*）等其他超参数，在所有实验中均保持不变。我们在 BRAWL 数据集上进行了全面的超参数搜索，测试了多种组合。相比之下，Cerberus Traces 数据集的规模大得多，所需运行时间也更长，因此我们只测试了一种超参数组合。

研究发现

本研究的主要目标是开发并验证一种稳定的深度强化学习环境，能够将复杂的 Sysmon 日志建模为图并进行分析。一份报告收集了各次运行中回合奖励总值的最小值和最大值，展示了该环境在训练过程中的稳定性和可靠性。另一份报告则跟踪并汇总训练指标：计算全局步数、回合数、平均运行奖励、平均运行策略损失、平均运行价值损失、平均运行熵，以及当前运行的奖励和步数。

运行指标通过汇总所有并行工作进程的数据取平均值。例如，平均策略损失的计算方法是将所有工作进程在该次运行中的策略损失总和除以总步数。策略损失衡量智能体的策略学习如何采取能够最大化奖励的动作。价值损失衡量评论家对某一状态的估计价值与环境中实际观测到的回报之间的差异。

Cerberus Traces 环境

使用 Cerberus Traces 数据集，我们观察到一次成功的实验运行，证明了深度强化学习环境的稳定性和功能。如图 4 所示，Cerberus 深度强化学习环境的价值损失在初期迅速下降，从较高的起始值急剧降低，随后在约 1200 个训练步数后稳定在波动但幅度较低的状态。这表明评论家网络成功学会了以合理的准确度预测高度稀疏且复杂的图状态的价值。我们的深度强化学习环境在 10 小时 42 分钟内完成了 4,710 个训练步数，展现出长期稳定性，以及支持强化学习智能体进行长时间训练和迭代训练的能力。

引入并行化后，不出所料地产生了内存瓶颈，其受模型输出维度数量和并行环境工作进程数量的影响。Cerberus Traces 数据集在两个工作进程且模型输出维度为 64 的情况下，每次运行能够完成 24 到 120 步，在未并行化时平均每次运行 537 秒。但在保持模型超参数不变的情况下，完成的步数有所下降，每次运行分别为 18、72 或 81 步，平均持续时间为 292 秒。所有使用 Cerberus Traces 网络的并行运行均因图形处理器内存不足错误而提前终止。

BRAWL 环境

表 3 表明，以更分散的节点交互为特征的 Cerberus Traces 深度强化学习环境产生了稳定且快速收敛的结果（损失标准差为零）。相反，BRAWL 环境尽管规模更小，却表现出极端的训练不稳定性（平均损失高且标准差高，例如1.01×10^{13}）。BRAWL 深度强化学习环境证实，其相对更稀疏的结构以及较低的节点度对强化学习智能体构成了一个根本更难的问题。尽管如此，在各种超参数组合和训练轮次中，BRAWL 环境始终产生有效观测，并管理回合状态转换。

在 BRAWL 深度强化学习环境中，依次完成了 542 次实验。在保持并行工

	BRAWL	Cerberus
平均奖励（均值）	-18.42	-4.66
平均片段（标准差）	63.83	0.76
平均策略损失（均值）	0.09	0
平均策略损失（标准差）	26.42	0
平均价值损失（均值）	1.21E+11	4.58E+08
平均价值损失（标准差）	1.01E+13	6.87E+08

表 3：深度强化学习环境的并行工作器训练指标平均值。Cerberus 环境的损失方差可以忽略不计，表明其具有可行性和稳定性。相比之下，BRAWL 中较大的损失标准差（1.01×10^{13}）证实，其稀疏图结构给深度强化学习智能体带来了严峻的稳定性问题。

在工作线程数量保持不变的情况下，各实验的运行时长为 1 至 3 秒，表明该环境在各种超参数配置下表现可靠且可预测。

结论

本文介绍了一种新方法，利用深度强化学习在由 Sysmon 日志生成的系统事件图上进行自动化安全分析和攻击路径映射。与依赖网络流或人工整理规则的现有安全研究不同，我们的工作直面系统事件的客观进程级分析问题。强化学习智能体的状态由节点（进程）明确定义，动作则是选择一个因果目标节点（s'）。至关重要的是，奖励经过了精心设计，并借鉴图分析中有关稀疏性和中心化进程的见解，引导智能体根据边有效性、进程属性（如完整性级别）和网络指标进行遍历，从而创建一个可行的环境，使智能体能够学习具体的系统攻击路径行为。

该框架重点关注三项主要挑战：平衡模型参数维度与处理效率；设计能够将稀疏的底层事件与安全结果关联起来的奖励函数；以及有策略地选择要建模的相关系统事件和用户操作。Cerberus Traces 数据集的稀疏性和可导航结构对智能体的探索至关重要，证明了我们设计的稳定性和可扩展性。

总之，这项工作证明了从操作系统日志构建图形化强化学习环境的可行性，并指出有效的奖励设计和有策略的事件选择是主要挑战。*SubstratumGraphEnv*是未来研究的重要平台。

附录和补充材料

以下方程用于分析 Cerberus Traces 和 BRAWL 数据集的网络结构。

边密度（ρ）对于有向图：

$$\rho = \frac{m}{(n(n-1))} \tag{6}$$

其中， m 是边的数量， n 是节点的数量。

聚类系数：子图边权重的地理平均值（c_u）：

$$c_u = \frac{1}{\deg(u)(\deg(u)-1)} \sum_{vw} \left(\hat{w}_{uv}\hat{w}_{uv}\hat{w}_{uv}\right)^{1/3} \tag{7}$$

用 $\hat{}w_{uv}$ 表示的边权重通过将原始权重 $\hat{}w_{uv}$ 除以网络中的最大权重 max(w)进行归一化。也就是说，$\hat{}w_{uv} = w_{uv}/\max(w)$。对于度小于 2（deg($u$) < 2）的任何节点 u，聚类系数 c_u 定义为 0。

平均度连通性 (k)：针对有向图：

$$k^{w}_{nn,i} = \frac{1}{s_i} \sum_{j \epsilon N(i)} w_{ij} k_j \tag{8}$$

节点的入度中心性 (c_{in})：

$$c_{in}(v) = \frac{\deg^{in}(v)}{n-1} \tag{9}$$

作者贡献

作者 1 概念化、方法论、分析、数据整理、写作。

作者 2 审阅与监督。**作者 3** 审阅与监督。

参考文献

Almasan, P.；Su´arez-Varela, J.；Rusek, K.；Barlet-Ros, P.；以及 Cabellos-Aparicio, A. 2019。深度强化学习遇见图神经网络：探索路由优化用例。出版方：arXiv 版本号：3。

Applebaum, A.；Miller, D.；Strom, B.；Korban, C.；以及 Wolf, R. 2016年。智能、自动化的红队仿真。载于第*32*届计算机安全应用年度会议论文集，363–373。美国加利福尼亚州洛杉矶：ACM。国际标准书号 978-1-4503-4771-6。

Bland, J. A.；Petty, M. D.；Whitaker, T. S.；Maxwell, K. P.；以及 Cantrell, W. A. 2020。机器学习网络攻击与防御策略。《计算机与安全》，92：101738。Botvinick, M.；Ritter, S.；Wang, J. X.；Kurth-Nelson, Z.；Blundell, C.；以及 Hassabis, D. 2019。强化学习，快与慢。《认知科学趋势》，23(5)：408–422。

C. David Hylender；Philippe Langlois；Alex Pint；和 Suzanne Widup。2025。《2025 年数据泄露调查报告》。技术报告，Verizon。

Cheng, N.; Wang, X.; Li, Z.; Yin, Z.; Luan, T. H.; and Shen, X. 2025. 借助数字孪生提升基于强化学习的资源管理：机遇、应用与挑战。《*IEEE*网络》, 39(1): 189–196.

Chowdhary, A.; Huang, D.; Mahendran, J. S.; Romo, D.; Deng, Y.; and Sabur, A. 2020. 自主安全分析与渗透测试。载于*2020*年第*16*届国际移动、感知与网络会议（*MSN*）, 508–515. 日本东京：IEEE. 国际标准书号 978-1-7281-9916-0。

Claypoole, J.; Cheung, S.; Gehani, A.; Yegneswaran, V.; and Ridley, A. 2025. 解读基于强化学习的网络战模拟平台中的智能体行为。arXiv:2506.08192 [cs]. Cody, T.; Rahman, A.; Redino, C.; Huang, L.; Clark, R.; Kakkar, A.; Kushwaha, D.; Park, P.; Beling, P.; and Bowen, E. 2022. 使用攻击图和强化学习发现数据渗出路径。arXiv:2201.12416 [cs]. Deokar, B.; and Hazarnis, A. 2012. 利用日志文件和强化学习实现入侵检测系统。《国际计算机应用期刊》, 45(19): 28–35. Edeh, V.; Korki Mehdi; and Mekhilef, Saad. 2024. 面向信息物理系统数据完整性攻击检测的强化学习赋能图卷积网络框架。《*CSEE*电力与能源系统期刊》. Elderman, R.; J. J. Pater, L.; S. Thie, A.; M. Drugan, M.; and M. Wiering, M. 2017. 网络安全模拟中的对抗性强化学习：。载于第*9*届智能体与人工智能国际会议论文集, 559–566. 葡萄牙波尔图：SCITEPRESS - 科学与技术出版物。国际标准书号 978-989-758-219-6 978-989-758-220-2. Ergun, S.; Sammour, I.; and Chalhoub, G. 2023. 综述网络模拟器如何为无线网络中的强化学习提供支持。《计算机网络》, 234: 109934. Farama Foundation. 2025a. 复合空间。Farama Foundation. 2025b. 基本空间。Farama Foundation. 2025c. gymnasium.spaces.graph 的源代码。Fritz Obermeyer. 2017. pytorch/torch/distributions/categor-ical.py at v2.7.0 · pytorch/pytorch. Gueriani, A.; Kheddar, H.; and Mazari, A. C. 2024. 物联网入侵检测的深度强化学习：综述。出版方：arXiv 版本号：1. Harries, L.; Clarke, R. S.; Chapman, T.; Nallamalli, S. V. P. L. N.; Ozgur, L.; Jain, S.; Leung, A.; Lim, S.; Dietrich, A.; Hern´andez-Lobato, J. M.; Ellis, T.; Zhang, C.; and Ciosek, K. 2020. DRIFT：面向功能性软件测试的深度强化学习。arXiv:2007.08220 [cs]. Ispahany, J.; Islam, M. R.; Khan, M. A.; and Islam, M. Z. 2025. 用于勒索软件分析与检测的 Sysmon 增量学习系统。arXiv:2501.01089 [cs]. Jahin, M. A.; Soudeep, S.; Farid, F. A.; Mridha, M. F.; Kabir, R.; Islam, M. R.; and Karim, H. A. 2025. CAGN-GAT 融合：用于网络入侵检测的混合式对比注意力图神经网络。arXiv:2503.00961 [cs]. Jamshidi, S.; Nikanjam, A.; Nafi, K. W.; Khomh, F.; and Rasta, R. 2025. 深度强化学习在物联网入侵检测中的应用：系统综述。物联网, 31: 101531. arXiv:2504.14436 [cs]. Jeff Hancock. 2022. 《人为错误心理学：理解危及贵公司网络安全的失误》。技术报告，Tessian Limited，斯坦福大学。Kenyon, A.; Deka, L.; 和 Elizondo, D. 2020. 公共入侵数据集是否适合用途：描述入侵事件数据集的最新进展。《计算机与安全》, 99: 102022. Kujanp¨a¨a, K.; Victor, W.; 和 Ilin, A. 2021. 使用深度强化学习实现权限提升自动化。载于第*14*届*ACM*人工智能与安全研讨会论文集, 157–168. 线上活动，大韩民国：ACM. ISBN 978-1-4503-8657-9. Kulkarni, P.; 和 OShaughnessy, S. 2025. 使用动态图神经网络检测恶意软件。欧洲网络战与安全会议, 24(1): 830–837. Li, Z.; Shi, J.; 和 Leeuwen, M. v. 2024. 基于图神经网络的日志异常检测与解释。arXiv:2307.00527 [cs]. Liu, H.; Su, W.; Li, T.; Huang, W.; 和 Li, Y. 2025. 数字孪生增强的多智能体强化学习，用于大规模移动网络覆盖优化。《*ACM*数据知识发现汇刊》, 19(1): 1–23. 洛斯阿拉莫斯国家实验室. 2024a. 平均度连通性 — NetworkX 3.4.2 文档。洛斯阿拉莫斯国家实验室. 2024b. 聚类 — NetworkX 3.4.2 文档。洛斯阿拉莫斯国家实验室. 2024c. 密度 — NetworkX 3.4.2 文档。洛斯阿拉莫斯国家实验室. 2024d. 入度中心性 — NetworkX 3.4.2 文档。L´opez-Montero, D.; ´Alvarez Aldana, J. L.; Morales-Mart´ınez, A.; Gil-L´opez, M.; 和 Garc´ıa, J. M. A. 2025. 用于自动化网络安全渗透测试的强化学习。版本号：1. Maeda, R.; 和 Mimura, M. 2021. 使用深度强化学习实现后利用自动化。《计算机与安全》, 100: 102108. Marbel, R.; Cohen, Y.; Dubin, R.; Dvir, A.; 和 Hajaj, C. 2024. 多云环境下异常频发：用于早期检测云服务用户异常的动态图神经网络。arXiv:2409.12726 [cs]. Mike Kemmerer; 和 Craig Wampler. 2018. mitre/brawl-public-game-001. 原始日期：2017-08-29T15:03:50Z. Mitra, S.; Chakraborty, T.; Neupane, S.; Piplai, A.; 和 Mittal, S. 2024. 图神经网络在辅助防御性网络行动中的应用。arXiv:2401.05680 [cs]. Mnih, V.; Badia, A. P.; Mirza, M.; Graves, A.; Lillicrap, T. P.; Harley, T.; Silver, D.; 和 Kavukcuoglu, K. 2016. 深度强化学习的异步方法。arXiv:1602.01783 [cs]. Molina-Markham, A.; Miniter, C.; Powell, B.; 和 Ridley, A. 2021. 面向自主网络防御的网络环境设计。arXiv:2103.07583 [cs]. Munikoti, S.; Agarwal, D.; Das, L.; Halappanavar, M.; 和 Natarajan, B. 2022. 图神经网络与深度强化学习的挑战与机遇：算法与应用综述。arXiv:2206.07922 [cs].

Pratomo, B. A. 2023. bazz-066/cerberus-trace. 原始日期：2023-07-06T11:17:46Z. Proofpoint, Inc. 2024. Proofpoint的《2024年CISO之声报告》显示，四分之三的首席信息安全官认为人为错误是主要网络安全风险。技术报告。PyTorch基金会. 2025. Linear — PyTorch 2.7 文档。Richard Barnes. 2021. torch.nn.functional.mse loss — PyTorch 2.7 文档。Sahu, A.; Venkatraman, V.; 和 Macwan, R. 2023. 面向网络弹性配电系统的强化学习环境。*IEEE Access*, 11: 127216–127228. SentinelOne. 2025. 50+2025年云安全统计数据。技术报告。Terranova, F.; Abdelkader Lahmadi; 和 Chrisment, I. 2024. 用于通信网络自动化网络攻击路径预测的深度强化学习。出版者：未发表。Toyama, D.; Hamel, P.; Gergely, A.; Comanici, G.; Glaese, A.; Ahmed, Z.; Jackson, T.; Mourad, S.; 和 Precup, D. 2021. AndroidEnv：面向 Android 的强化学习平台。arXiv:2105.13231 [cs]. Vincent Moens. 2024. [BUG] TorchRL 与 Gymnasium 1.0 不兼容：自动重置功能破坏模块化和数据完整性。Wang, C.; Zheng, P.; Gui, J.; Hua, C.; 和 Hassan, W. U. 2025. 我们达到目标了吗？揭示最先进的图网络入侵检测系统。arXiv:2503.20281 [cs]. Xie, Y.; Zhang, H.; 和 Babar, M. A. 2022. LogGD：利用图神经网络检测系统日志中的异常。arXiv:2209.07869 [cs]. Xu, X.; 和 Xie, T. 2005年. 使用系统调用序列进行基于主机的入侵检测的强化学习方法。载于 Hutchison, D.; Kanade, T.; Kittler, J.; Kleinberg, J. M.; Mattern, F.; Mitchell, J. C.; Naor, M.; Nierstrasz, O.; Pandu Rangan, C.; Steffen, B.; Sudan, M.; Terzopoulos, D.; Tygar, D.; Vardi, M. Y.; Weikum, G.; Huang, D.-S.; Zhang, X.-P.; 和 Huang, G.-B. 编, 智能计算进展, 第3644卷, 995–1003. 柏林，海德堡：Springer Berlin Heidelberg. ISBN 978-3-540-28226-6 978-3-540-31902-3. 丛书名称：计算机科学讲义。Yang, Y.; 和 Liu, X. 2022. 行为多样化自动渗透测试：一种好奇心驱动的多目标深度强化学习方法。arXiv:2202.10630 [cs]. Zhao, W.; Queralta, J. P.; 和 Westerlund, T. 2020. 深度强化学习中的仿真到现实迁移：综述。载于 *2020*年*IEEE*计算智能研讨会系列（*SSCI*）, 737–744. 澳大利亚首都领地堪培拉，澳大利亚：IEEE. ISBN 978-1-7281-2547-3. Zheng, W.; Wang, D.; 和 Song, F. 2021. OpenGraphGym-MG：使用强化学习在多GPU系统上解决大规模图优化问题。arXiv:2105.08764 [cs].