攻击技战术知识驱动的APT攻击路径推理方法

吕明琪[1,2]，盛起[1]，陈铁明[1]，朱添田[1]，王飞[3]
（1.浙江工业大学地理信息学院，浙江 杭州 310023；2.湖州工业控制技术研究院，浙江 湖州 313098；
3.中国石油大学（华东）控制科学与工程学院，山东 青岛 266580）

摘要：现有基于溯源图的高级持续性威胁（APT）攻击检测方法主要集中于单点攻击事件的检测，难以刻画多阶段攻击事件之间的时序关联与因果依赖关系，为此，围绕APT攻击路径推理问题（即将属于同一APT攻击的相关攻击事件聚合为完整的攻击链），提出了一种攻击技战术知识驱动的APT攻击路径推理方法。该方法首先通过异常节点检测、攻击技战术识别与图精简构建包含孤立攻击事件的异常子图；随后引入基于威胁情报构建的ATT&CK攻击技战术序列模式，以指导攻击路径推理；最后结合图搜索与威胁评分机制，实现APT攻击链的重建。在模拟攻击采集的内核日志数据集以及公开的DARPA TC数据集上的实验结果表明，所提方法在保证攻击链完整性的前提下，重建精确度较现有方法提高60%以上。

关键词：高级持续性威胁；溯源图；异常检测；攻击路径推理；ATT&CK攻击技战术
中图分类号：TP309
文献标志码：A
doi：10.11959/j.issn.1000–436x.TXXB260092

Attack tactics and techniques knowledge-driven APT attack path reasoning method

Lyu Mingqi[1,2], Sheng Qi[1], Chen Tieming[1], Zhu Tiantian[1], Wang Fei[3]
1. College of Geoinformatics, Zhejiang University of Technology, Hangzhou 310023, China
2. Huzhou Industrial Control Technology Research Institute, Huzhou 313098, China
3. College of Control Science and Engineering, China University of Petroleum (East China), Qingdao 266580, China

Abstract: Existing provenance graph-based advanced persistent threat (APT) attack detection methods mainly focues on identifying isolated attack events and fail to capture the temporal correlations and causal dependencies among multi-stage attack events. To address this issue, the problem of APT attack path reasoning was investigated, which aimed to aggregate related attack events belonging to the same APT campaign into a complete attack chain, and an attack tactics and techniques knowledge-driven APT attack path reasoning method was proposed. Specifically, the proposed method first constructed an anomaly subgraph containing isolated attack events through anomaly detection, attack tactics and techniques identification, and graph pruning, then introduced an ATT&CK-based tactic-technique sequence pattern built from threat intelligence to guide the attack path reasoning, and finally reconstructed complete APT attack chains by integrating graph search with a threat scoring mechanism. Experimental results on a simulated attack dataset collected from kernel logs and the public DARPA TC dataset demonstrate that under the premise of maintaining attack chain integrity, the proposed method improves the reconstruction precision by over 60% compared with existing methods.

Key words: advanced persistent threat, provenance graph, anomaly detection, attack path reasoning, ATT&CK attack tactics and techniques

收稿日期：2026–02–09；**修回日期：**2026–03–31
通信作者：陈铁明，tmchen@zjut.edu.cn
基金项目：国家自然科学基金资助项目（No.62372410, No.U22B2028）；浙江省“尖兵”科技计划基金资助项目（No.2025C01013, No.2024C01066）；杭州市重点研发计划基金资助项目（No.2024SZD0220）；湖州市重点研发计划基金资助项目（No.2025ZD2037）；绍兴市重点研发计划基金资助项目（No.2025B11004）
Foundation Items: The National Natural Science Foundation of China (No.62372410, No.U22B2028), The Zhejiang Province Leading Goose Program (No. 2025C01013, No. 2024C01066), The Key Research Program of Hangzhou (No. 2024SZD0220), The Key Research Program of Huzhou (No.2025ZD2037), The Key Research Program of Shaoxing (No.2025B11004)

0 引言

高级持续性威胁（advanced persistent threat, APT）是一种高度隐蔽、长期持续的复杂网络攻击，攻击者通过多阶段渗透过程窃取敏感数据或破坏关键基础设施[1]。随着攻击手段日趋复杂，传统安全防御机制难以有效应对此类威胁，给各类企业及机构带来严重安全风险。为应对这一挑战，现有研究引入溯源分析技术增强APT攻击监测能力。该技术通过将操作系统内核日志数据建模为一种以系统实体（如进程、文件）及其交互事件（如创建进程、读取文件）为节点和边的图结构（称为溯源图），构建系统活动的全局上下文关联视图，为深入分析APT攻击提供支持[2-5]。然而，现有方法大多局限于识别孤立攻击事件，而一次APT攻击通常由多个步骤的长期攻击行为组成，导致孤立攻击事件难以还原完整的攻击链。这一局限性不仅导致大量误报，更使安全分析人员难以全面评估APT攻击范围与影响，阻碍了防御策略的快速制定。

一次APT攻击通常会产生多个攻击事件（如不同的攻击技战术产生的攻击事件），而攻击路径推理指将分散的独立攻击事件聚合成完整的攻击链，从而揭示攻击事件之间的因果关联，是APT攻击诊断的重要辅助手段[6]。与传统的系统日志数据不同，溯源图由操作系统底层的内核日志数据构建得到，因此规模过于庞大且依赖关系复杂，导致人工分析技术难以适用。近年来，面向溯源图的APT攻击路径推理方法主要分为3种。（1）基于异常检测的方法：通过检测和保留偏离正常行为模式的节点或边，以压缩溯源图的规模，从而实现高效的攻击路径推理分析[7]。然而，该方法仅能对节点和边进行恶意/良性的二元分类，无法区分哪些节点和边属于同一攻击事件，难以适应攻击链交叠的复杂场景。（2）基于威胁情报的方法：将攻击路径推理转化为溯源图与威胁情报之间的模式匹配问题，尽管在已知攻击检测中效果良好，但由于威胁情报仅涵盖已知攻击，其应对零日攻击与未知威胁的能力受限[8]。（3）基于标签传播的方法：依赖人工设计的规则进行攻击路径推理，具备较好的灵活性与可解释性。但规则对环境与攻击变体较为敏感，泛化能力不足[5,9]。

此外，对大量APT攻击实例的分析表明，尽管攻击手段不断演进，但APT攻击在技战术层面仍呈现出一定的规律性，即不同攻击往往遵循相似的战术和技术阶段进行推进[10]。例如，具有类似目标的勒索软件攻击通常在“初始访问（Initial Access）→防御规避（Defense Evasion）→横向移动（Lateral Movement）→影响（Impact）”等阶段表现出规律性的战术序列模式[11-12]。此类模式的复现性揭示了APT攻击的结构化和系统化特征，为多阶段攻击事件关联提供了可指导的先验知识。

基于该关键洞察，本文提出了一种融合攻击技战术的APT攻击路径推理方法，该方法通过挖掘并利用ATT&CK攻击技战术序列模式，实现在攻击行为语义下对孤立攻击事件进行关联分析。首先，鉴于直接在大规模溯源图上进行推理的计算成本过高，采用节点级异常检测与图精简策略将溯源图压缩为仅保留可疑节点的异常子图，以降低溯源图的复杂度。其次，对异常子图中的可疑节点进行技战术标注，将二元可疑事件映射为高层攻击行为语义，为后续推理提供领域知识支撑。最后，挖掘ATT&CK对齐的技战术序列模式并设计威胁评分与路径评分机制，指导异常子图中的攻击路径推理与合并，从而实现攻击链重建。

本文的主要贡献如下。

（1）提出了一种攻击技战术知识驱动的APT攻击路径推理方法，将ATT&CK对齐的技战术序列模式引入溯源图路径推理过程。

（2）设计了融合技战术转换、节点邻近结构与异常分数等的多因素评分机制，用于指导异常子图中攻击路径的推理与筛选。

（3）在课题组模拟攻击数据集和公开数据集上进行了系统实验评估，结果表明本文方法在保持攻击链完整性的同时，有效提升了攻击链重建的精确性。

1 背景与动机

1.1 溯源图

由于内核日志数据体量巨大，基于内核日志数据的APT攻击分析面临挑战。为解决该问题，当前研究将内核日志数据组织为溯源图，通过上下文将潜在的攻击事件关联起来。溯源图是一种有向无环图，可定义为$G=(E,V)$，其中，V表示系统实体集合（如文件、进程、套接字），E表示系统实

体之间的交互关系集合（如读取文件、创建进程）。每条边 $e=(u,v,a,t)$ 记录一个系统事件，其中 u 为主体，v 为客体，a 为交互类型，t 为时间戳。通过构建溯源图，安全分析人员能够通过向后追溯有效定位入侵点，并通过向前追溯发现潜在影响，从而显著提升对APT攻击的推理与分析效率。本文构建的溯源图中主要的系统实体与交互类型如表1所示。

表1　　主要的系统实体与交互类型

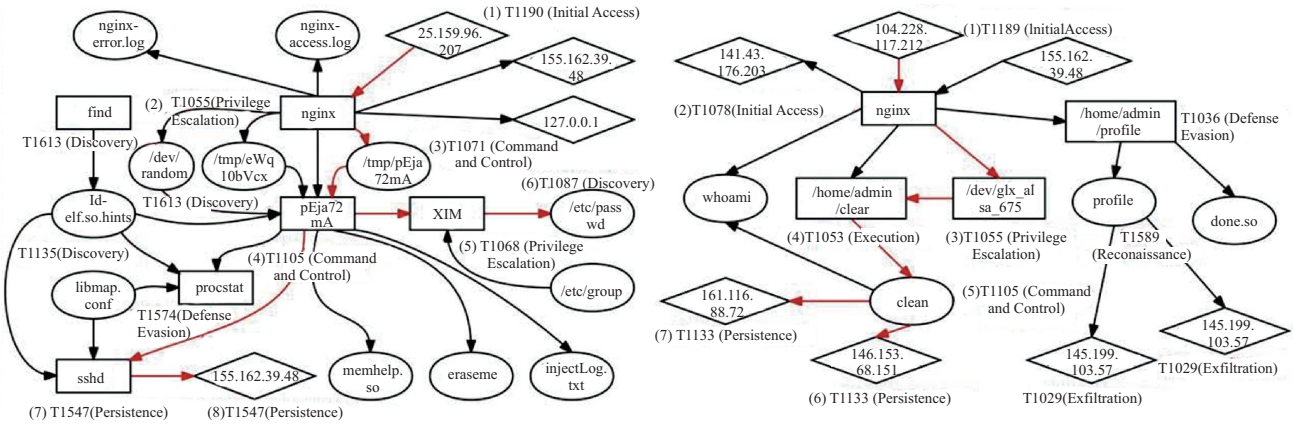
系统实体	交互类型	实体属性
Process→File	read, write, chmod	PID, Name, Path, Cmd
Process→Process	start, end, execve, clone	PID, Name, Cmd
Process→Socket	sendto, recvfrom, copy	PID, Name, Cmd, IP

1.2　研究动机

以DARPA TC Engagement 3[13]中的APT攻击实例为研究动机的攻击示例，图1展示了两个具有相似目标的APT攻击。（1）攻击1（Nginx漏洞利用攻击）：攻击者利用Nginx中的HTTP请求漏洞对CADETS系统发起攻击。该漏洞利用使drakon植入程序在Nginx进程内存中得以运行，并通过HTTP连接到远程控制台。攻击者随后将drakon可执行文件下载至目标磁盘，通过权限提升技术以root权限执行，最终建立具有最高权限的持久化后门，维持与C2服务器的通信。（2）攻击2（恶意广告驱动攻击）：攻击者通过网站上的恶意广告服务器发起攻击。当受害者访问网站时，该漏洞利用使drakon植入程序在Firefox进程内存中得以运行，并向外连接到攻击者控制台。攻击者将植入程序写入目标磁盘，通过权限提升技术以root权限执行，建立持久化机制以便在受害者重新访问网站时重新获取访问权限，部署附加攻击载荷，并维持开放的C2通道以进行持续渗透。尽管两次攻击利用的初始入口不同，但其后续攻击阶段呈现出高度相似的行为序列。

为便于分析，图1显示了对攻击精简后的异常子图，图1中节点代表检测到的异常实体，并标注了相应的ATT&CK攻击战术/技术，其中虚线箭头表示攻击路径。通过对比分析，识别出两个攻击中存在的一致战术/技术序列模式。具体而言，攻击1遵循的攻击路径为“T1190(Initial Access)→T1055 (Privilege Escalation)→T1071 (Command and Control)→T1105 (Command and Control)→T1068 (Privilege Escalation)→T1087 (Discovery)”；攻击2遵循的攻击路径为“T1189 (Initial Access)→T1078 (Initial Access)→T1055 (Privilege Escalation)→T1053 (Execution)→T1105 (Command and Control)→T1133 (Persistence)”。

通过分析上述攻击，可以得到以下两点结论。（1）攻击路径的结构化特征：两个攻击在战术层面共享相似的子序列，表明攻击者通常遵循具有阶段递进关系的技战术序列来实施攻击，这种结构化特征揭示了真实攻击路径在战术层面具有相对稳定的演化规律，为攻击路径的识别与区分提供了重要依据。（2）噪声节点的无序性特征：攻击路径之外的实体往往表现为不连贯的攻击战术子序列，缺乏明确的逻辑递进关系，整体呈现出较强的随机性。在复杂溯源图中，仅依赖局部结构信息或事件级异常



(a) 攻击1　　(b) 攻击2

图1　研究动机的攻击示例

检测结果，难以有效利用上述差异对噪声事件进行筛除。上述分析表明，在攻击路径推理过程中，有必要引入能够刻画攻击行为阶段演化特征的序列信息，以区分真实攻击路径与无关噪声事件。基于此，本文通过挖掘攻击战术与技术的序列模式，对溯源图中的攻击事件进行筛选，并利用攻击技战术转换关系的概率对攻击行为顺序的合理性进行评估，从而刻画真实的攻击路径，为APT攻击路径推理提供有效指导。

2 方法设计

2.1 框架概述

图2展示了本文方法的三阶段架构：异常子图挖掘、攻击路径推理和攻击路径评分与合并。各阶段概述如下。

（1）异常子图挖掘：首先，采用无监督异常检测算法识别溯源图中的异常节点；其次，识别异常节点的ATT&CK攻击技战术类型；最后，基于异常节点对原始溯源图进行剪枝处理，生成精简的异常子图。

（2）攻击路径推理：首先，通过挖掘带有攻击技战术类型标注的威胁情报，提取其中蕴含的攻击战术与技术序列模式；随后，基于提取的序列模式，应用图搜索算法在异常子图中识别潜在的候选攻击路径。

（3）攻击路径评分与合并：首先，综合考虑多种威胁因素计算各候选攻击路径的置信度评分，评估因素包括路径与攻击战术与技术序列模式的匹配程度、节点连通性及其潜在影响力、异常检测模块赋予节点的异常分数等；随后，对具有共同节点的候选攻击路径进行合并，以降低整体冗余度。

2.2 异常子图抽取

2.2.1 异常节点检测

由于内核日志数据的细粒度特性，溯源图的规模非常庞大，直接进行威胁检测与分析十分困难。为此，采用异常节点检测方法，定位可疑节点及其相关系统实体交互，过滤无关系统实体，以降低溯源图的规模与复杂度。由于溯源图中的异常节点隐藏在海量良性节点中，对每个节点进行人工标注以构建训练集在实际场景中可行度不高，难以采用有监督学习算法训练分类模型[14]。为此，遵循现有异常检测的思路[6,15-16]，通过无监督学习算法构建异常节点检测模型，即模型仅在良性数据集上训练，并基于节点行为与正常模式的偏离程度来识别异常。

具体而言，首先将溯源图中每个节点的特征初始化为20维特征向量，表示其所有入边和出边类型的频率统计（如表1所示）。此外，由于初始特征向量难以有效捕捉节点间的长距离因果依赖关系，引入图神经网络（graph neural network, GNN）进一步聚合节点的高阶交互特征。每个GNN层负责融合其相邻节点的信息，通过堆叠多个GNN层，模型能够学习更远距离节点间的交互语义。最终得到每个节点的语义嵌入向量，如式(1)所示。

$$E^{(t)} = \mathrm{GNN}\left(W^{(t-1)}, G, E^{(t-1)}\right) \tag{1}$$

基于得到的节点的语义嵌入向量，采用iForest算法进行异常检测。该算法通过随机特征选择机制增强对关键特征的敏感性。最终，所有节点均获得相应的异常分数，将其中异常分数高于预设阈值的节点判定为异常节点。

2.2.2 节点攻击技战术识别

通过识别每个异常节点所属的ATT&CK攻击技战术类型，可以将相关攻击事件映射到

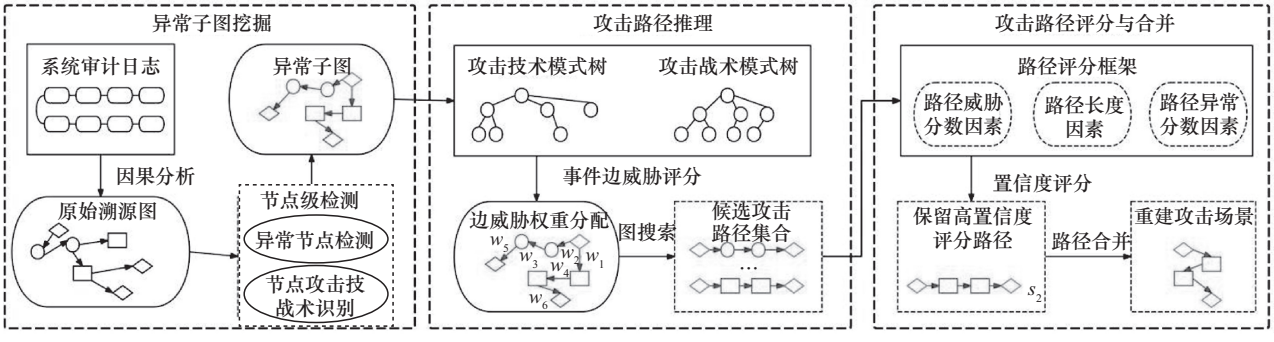


图2 本文方法框架

ATT&CK攻击框架中的具体战术与技术条目，为后续攻击技战术驱动的APT攻击路径推理分析提供支持。本文采用文献[15]实现节点级的攻击技战术识别。具体而言，首先从异常节点中筛选具有高度连通性的节点作为种子节点，通过深度优先搜索构建技术子图，并使用图嵌入技术将其编码为低维向量。随后，采用基于孪生网络的小样本学习策略训练攻击技战术识别模型，通过在特征空间中进行距离度量，实现在每类攻击技战术样本数量有限的条件下对攻击技战术的有效识别。所有异常节点均按此流程处理，以完成相应攻击技战术的类型标注。

2.2.3　溯源图精简

在发现溯源图中的所有异常节点之后，就可以根据异常节点对溯源图进行精简，具体方法如下。首先，筛选所有不存在前置异常节点的异常节点EN_k，并将其初始化为入口节点集合（entry node set, ENS）。针对ENS中的每个异常节点EN_k，沿依赖边执行深度优先搜索以枚举所有原始路径。在逐个遍历源自异常节点EN_k的原始路径时，路径上的节点被迭代处理，直至遇到后续异常节点AN_{k+1}。此时，非异常中间节点将被剪枝处理，同时保留有向边$AN_k \rightarrow AN_{k+1}$。该精简流程递归执行，直至当前路径无法继续抵达后续异常节点。通过这种迭代式精简，最终生成的异常子图仅保留核心攻击行为的拓扑结构。

图3展示了一个具体示例，原始溯源图经过异常节点检测，节点被分类为异常节点（以虚线表示）或良性节点（以实线表示）。首先，选取不存在前置异常节点的异常节点1和异常节点2作为深度优先搜索的入口节点，生成7条原始路径：p_1 = (1) → (3) → (8) → (11) → (14)、p_2 = (1) → (4) → (9) → (11) → (14)、p_3 = (1) → (4) → (9) → (6) → (10) → (12) → (15)、p_4 = (1) → (4) → (9) → (6) → (10) → (13) → (16)、p_5 = (2) → (6) → (10) → (12) → (15)、p_6 = (2) → (6) → (10) → (13) → (16)和p_7 = (2) → (5)。随后，通过对非异常中间节点进行剪枝处理，这些原始路径被压缩为：p'_1 = (1) → (8) → (11) → (14)、p'_2 = (1) → (9) → (11) → (14)、p'_3 = (1) → (9) → (6)、p'_4 = (1) → (9) → (6) → (16)、p'_5 = (2) → (6)、p'_6 = (2) → (6) → (16)和p'_7 = (2)，最终生成包含节点{(1), (2), (6), (8), (9), (11), (14), (16)}的异常子图。该过程在过滤大量无关节点的同时，保留了潜在的异常传播路径，在保持攻击路径拓扑完整性的基础上显著降低了溯源图的复杂度。

2.3　攻击路径推理

2.3.1　攻击技战术模式挖掘

如第2.2节所述，攻击技战术模式能够有效关联溯源图中孤立的异常节点，并通过过滤偏离预期攻击行为的无关节点，辅助构建合理的攻击路径。通过以下两个步骤挖掘攻击技战术模式：首先，收集带有ATT&CK攻击技战术类型标注的威胁情报，并抽取攻击技术序列数据集；其次，通过挖掘收集的攻击技术序列数据集，构建攻击技战术序列模式树（以下简称ATT-SPT）。

如图4所示，第一阶段主要从AlienVault OTX、MISP、VirusTotal等开源威胁情报平台系统性地收集公开可用的威胁情报。原始威胁情报以HTML或PDF格式下载并归档存储。针对每份威胁情报，首先使用正则表达式直接从文本中提取ATT&CK攻击技术标识符（标识符的格式为“Txxxx”，其中xxxx代表数字编号）。对于未明确标注攻击技术的威胁情报，则通过人工提取其中与特定攻击技术

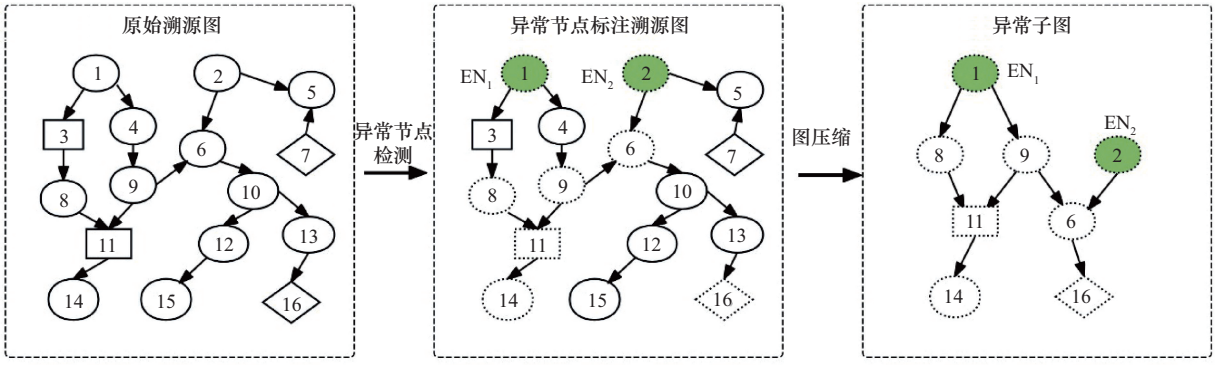


图3　溯源图精简示例

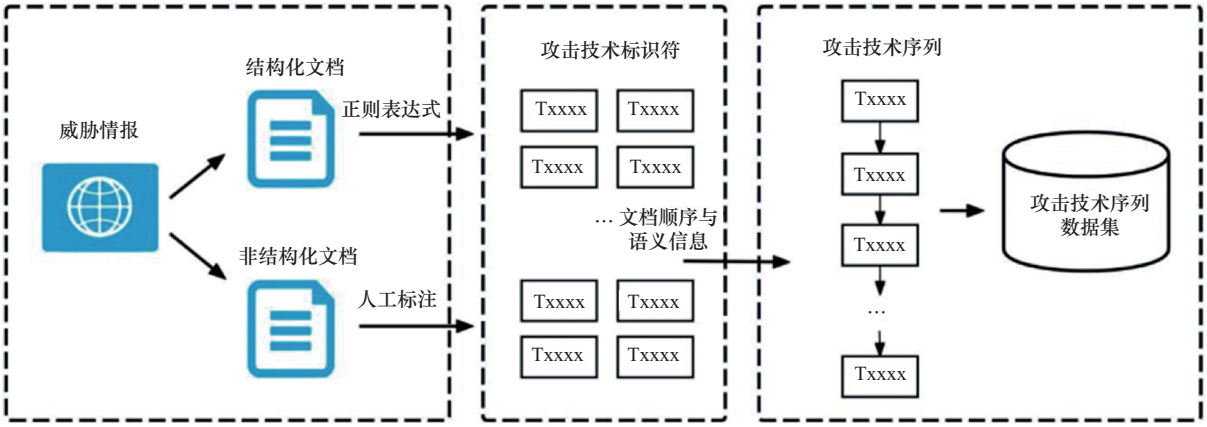


图4　攻击技术序列数据集构建过程

相关的关键术语，并将其映射至相应的ATT&CK攻击技术标识符[10]。随后，根据攻击技术在报告中出现的先后顺序，将提取的攻击技术标识符生成有向的攻击技术序列，所有生成的序列被归档形成攻击技术序列数据集。

在第二阶段，基于前述攻击技术序列数据集逐步构建ATT-SPT，由攻击技术模式树与攻击战术模式树两部分组成，其中攻击战术模式树通过攻击技术模式树映射推导获得。对攻击技术模式树的构建而言，从攻击技术序列数据集中挖掘攻击技术序列模式，并将其组织为树型结构。具体步骤如下：首先创建根节点以初始化攻击技术模式树；随后对攻击技术序列数据集中的每个攻击技术序列ts_k，按以下方式迭代扩展该树结构。

（1）若ts_k的前缀与攻击技术模式树中的任何分支均不匹配，则将ts_k作为新分支插入根节点之下。

（2）若ts_k的任一前缀与现有分支的前缀匹配，则将ts_k的不匹配部分插入该分支前缀的末尾。

攻击技术模式树通过对每个攻击技术序列迭代执行上述扩展过程而逐步构建。以图5为例，假设攻击技术序列数据集包含4个攻击技术序列，即ts_1 = T1589 → T1566 → T1059 → T1140 → T1105、ts_2 = T1584 → T1190 → T1505 → T1056 → T1071、ts_3 = T1583 → T1190 → T1090，以及ts_4 = T1583 → T1190 → T1059 → T1070 → T1056 → T1087，攻击技术模式树经过4次迭代后完成构建。最后，通过将攻击技术模式树中每个节点的攻击技术序列映射至其对应的攻击战术，并过滤连续重复的攻击战术节点，可从攻击技术模式树转换得到相应的攻击战术模式树。

2.3.2　异常子图边威胁评分

本节基于ATT-SPT对异常子图中的每条有向边进行威胁评分。具体而言，连接两个异常节点的每条有向边不仅对应系统事件的抽象表示，更代表一次攻击技战术的转换，反映了攻击者意图的演进。例如，从技术T1556（凭据访问）转换至T1570（横向移动），表明攻击者意图从获取凭据转向在网络

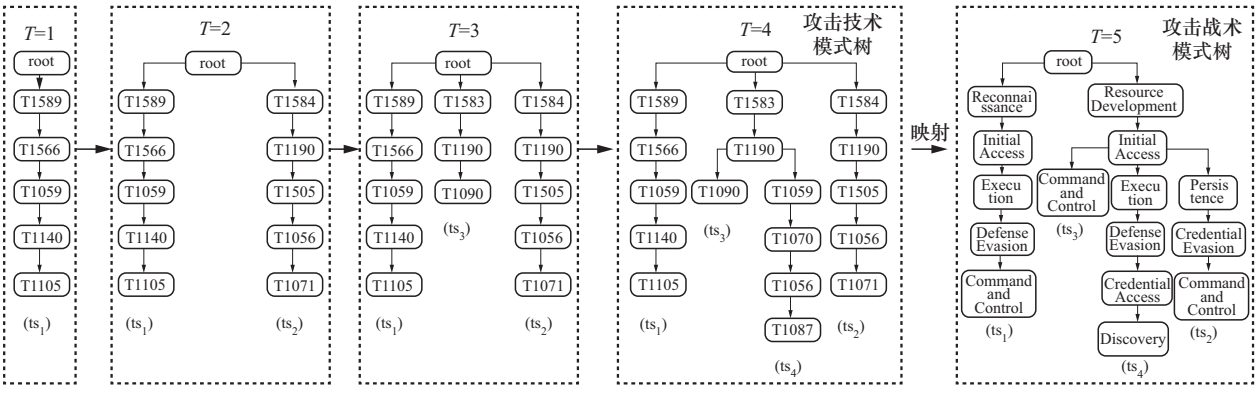


图5　ATT-SPT构建示例

内部扩展访问范围。这种转换通常发生在攻击者利用窃取凭据成功定位目标数据，并试图渗透其他系统以扩大危害的场景中。因此，可以通过分析攻击技战术转换与ATT-SPT中攻击技战术序列模式的匹配程度来评估其威胁程度。最终，从以下3个维度综合计算异常子图边威胁分数。

（1）攻击战术威胁分数评估：旨在量化攻击战术转换的威胁程度。给定一条代表攻击战术转换$\mathrm{TA}_s \to \mathrm{TA}_t$的边$e_k$（如初始访问→权限提升），该评分通过分析攻击战术模式树中TA_s与TA_t之间所有可达路径得出，包括直接和间接可达路径。在评分策略中，对于直接可达路径赋予较高的权重，同时根据间接可达路径长度按比例降低其权重，因为较长的中间序列通常意味着战术转换的连贯性较低，因而威胁关联性也相应减弱。特别地，同一战术间的转换被视为战术延续（即$\mathrm{TA}_s=\mathrm{TA}_t$），评分策略同样对其赋予最大权重。攻击战术威胁评分$\mathrm{TacticScore}(e_k)$按式(2)计算，其中$P(\mathrm{TA}_s,\mathrm{TA}_t)$表示攻击战术模式树中从$\mathrm{TA}_s$到$\mathrm{TA}_t$的所有路径集合。对于每条路径$p \in P(\mathrm{TA}_s,\mathrm{TA}_t)$，其拓扑长度量化为$\mathrm{len}(p)$，$\mathrm{out}(\mathrm{TA}_s)$表示从攻击战术$\mathrm{TA}_s$出发的所有分支路径数量。最后，使用标准化度量参数$b$调节评分的区分度。

$$\mathrm{TacticScore}(e_k)=\sum_{p\in P(\mathrm{TA}_s,\mathrm{TA}_t)}\frac{b}{\mathrm{out}(\mathrm{TA}_s)\mathrm{len}(p)} \tag{2}$$

（2）攻击技术威胁分数评估：攻击技术转换可视为攻击战术转换的具体实现手段。当攻击者在不同攻击战术阶段推进时，通常会从多种攻击技术中选择合适手段以实现攻击目标。即便在同一攻击战术阶段内，多种攻击技术也可能被协同编排以执行复杂的攻击操作。与受高层攻击逻辑约束的攻击战术转换不同，攻击技术转换具有更强的不确定性。因此，为量化攻击技术转换的威胁程度，重点关注该转换是否存在于攻击技术模式树中。若攻击技术模式树中两个攻击技术之间存在可达路径，则认为其构成潜在威胁。根据路径类型的不同，为直接可达路径分配较高威胁分数，为间接可达路径分配较低威胁分数，为攻击技术模式树中不存在可达路径的情况分配最小基础分数。形式上，对于给定的攻击技术转换$\mathrm{TA}_s \to \mathrm{TA}_t$，攻击技术威胁评分$\mathrm{TechniqueScore}(e_k)$按式(3)计算，其中$\Pi_{\mathrm{direct}}$是指示函数，若$\mathrm{TA}_s \to \mathrm{TA}_t$存在于攻击技术模式树中则为1，否则为0。$P_i$表示从$\mathrm{TA}_s$到$\mathrm{TA}_t$的所有间接路径集合。对于每条路径$p \in P_i$，其拓扑长度量化为$\mathrm{len}(p)$。此外，$\varepsilon$是未观测到转换设置的最小基础分数。最后，使用标准化度量参数a调节评分的区分度。

$$\mathrm{TechniqueScore}(e_k)=a\max\left\{\varepsilon,\Pi_{\mathrm{direct}},\max_{p\in P_i(\mathrm{TA}_s,\mathrm{TA}_t)}\frac{1}{\mathrm{len}(p)}\right\} \tag{3}$$

（3）邻近关系评估：除攻击技战术层面的转换特征外，边$e_k=(u,v)$的威胁评分还受其两端节点与其他邻居节点交互关系的影响，尤其体现在其入度和出度的结构上[17]。具体如下。

① 若节点u的入度远大于其出度，则表明u受大量其他节点的影响，且与节点v存在信息的集中交互。

② 若节点v的出度远大于其入度，则表明v高度依赖于u，且对其他节点具有广泛影响。

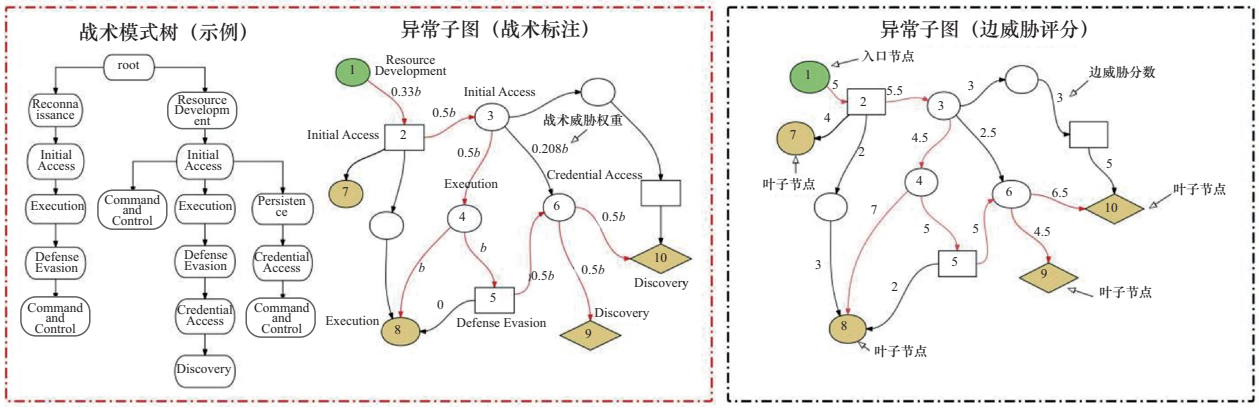
基于上述观察，使用式(4)计算边e_k的邻近关系威胁评分，记为$\mathrm{ScoreNI}(e_k)$。

$$\mathrm{ScoreNI}(e_k)=\frac{u.\mathrm{indegree}}{u.\mathrm{outdegree}}+\frac{v.\mathrm{outdegree}}{v.\mathrm{indegree}} \tag{4}$$

边e_k的最终威胁评分按式(5)定义为上述3类评分的加权和。

$$\mathrm{ScoreE}(e_k)=\alpha\mathrm{TacticScore}(e_k)+\beta\mathrm{TechniqueScore}(e_k)+(1-\alpha-\beta)\mathrm{ScoreNI}(e_k) \tag{5}$$

图6(a)给出了一个异常子图边威胁评分机制的示例。每条边的攻击战术转换评分基于其在攻击战术模式树中的可达路径计算。以节点3（初始访问）到节点6（凭据访问）的边为例，其评分流程如下：在攻击战术模式树（示例）中，这两个攻击战术间存在长度分别为3和2的间接可达路径，且从初始访问攻击战术节点出发有4个分支序列，由式(2)计算得分为$\frac{1}{4}\left(\frac{1}{3}+\frac{1}{2}\right)b=0.208\,3b$。该评分流程被迭代应用于计算异常子图中所有边的攻击战术转换评分。关键评分结果表明，对于节点4（执行）和节点8（执行）之间的攻击战术转换（属于同一攻击战术），因其代表攻击战术的延续获得了最高分b。相反，节点5（防御规避）到节点8（执行）的边在攻击战术模式树中不存在有效路径，导致评分为0。节点5（防御规避）到节点6（凭据访问）的边则因存在直接转换路径，且从防御规避节点出发



(a) 战术威胁评分示例　　(b) 路径回溯推理示例

图6　攻击路径推理示例

的分支数较少（总共两条分支中的一条即直接转换路径），获得0.5*b*的较高评分。该结构特征表明，在历史攻击案例中，从防御规避到凭据访问的转换具有较高的发生概率，而相较于节点3到节点6通过多跳间接路径实现的战术转换，节点5到节点6的战术直接转换更具威胁性，因此得到了更高的威胁评分。

2.3.3　攻击路径遍历

在完成异常子图边威胁评分后，通过遍历异常子图提取候选攻击路径。具体而言，采用一种逆向图遍历算法，该算法从出度为0的叶子节点出发，沿有向边逆向回溯。在每个回溯过程中，选择当前威胁评分最高的边，直至回溯到入度为0的入口节点。由此产生的节点序列构成一条候选攻击路径。图6(b)展示了这一过程的具体示例，基于编号为7~10的叶子节点，通过沿当前威胁评分最高的边向前回溯，获得4条候选攻击路径，即(1) →(2) →(7)、(1) →(2) →(3) →(4) →(8)、(1) →(2) →(3) →(4) →(5) →(6) →(9)和(1) →(2) →(3) →(4) →(5) →(6) →(10)。分析可见，推理得到的候选攻击路径覆盖了真实攻击路径(1) →(2) →(3) →(4) →(5) →(6) →(8) →(9) →(10)，但引入了部分误报节点，该问题将在2.4节中解决。

2.4　攻击路径评分与合并

为每条候选攻击路径分配一个置信度评分，旨在保留高置信度路径，并将具有共同节点的冗余路径合并，从而构建覆盖完整攻击链的攻击链路。具体而言，给定一条候选攻击路径CP，从以下多个维度综合量化攻击路径的置信度评分。

（1）边威胁分数因素：根据式(6)计算候选攻击路径中边威胁评分的平均值，以评估其局部威胁程度。

$$\mathrm{EdgeScore}(\mathrm{CP})=\frac{\sum_{e_k\in \mathrm{CP.edges}}\mathrm{ScoreE}(e_k)}{\mathrm{EdgeNumber}} \tag{6}$$

（2）路径长度因素：旨在过滤两类异常路径，一类是长度过短但具有高平均威胁评分的路径，另一类是由大量低威胁分数边组成的、可能获得虚高分的长攻击路径。路径长度因素的计算如式(7)所示。

$$\mathrm{LengthScore}(\mathrm{CP})=\left(1-\mathrm{e}^{-\frac{\mathrm{len}(\mathrm{CP})}{\lambda}}\right)\mathrm{EdgeScore}(\mathrm{CP}) \tag{7}$$

其中，参数λ可调整以优化对路径长度的敏感程度。

（3）节点异常分数因素：综合考虑候选攻击路径中每个异常节点的异常检测评分。由于3.2.1节中的异常节点检测模型会为每个节点分配一个异常分数，因此计算路径中所有节点异常分数的平均值，以保留具有高平均异常值的路径作为潜在的真实攻击路径。

$$\mathrm{NodeScore}(\mathrm{CP})=\frac{\sum_{v\in \mathrm{CP.Nodes}}\mathrm{AbnormalScore}(v)}{\mathrm{NodeNumber}} \tag{8}$$

最终，通过以下步骤重建完整的攻击链：首先，基于式(9)以加权求和的形式计算每条候选攻击路径CP的置信度评分；其次，保留置信度评分高于预设置信度阈值θ的候选攻击路径；最后，根据节点标识符（如进程ID等）识别不同路径间的共同节点，并基于这些共同节点将具有重叠结构的路径进行合并。对于与任何其他路径均无共同节

点的孤立路径，将被视为低置信度路径并被过滤。图7展示了高置信度攻击路径评分与合并的具体示例。候选攻击路径集包含4条初始路径：$path_1$ = (1) → (2) → (7)、$path_2$ = (1) → (2) → (3) → (4) → (8)、$path_3$ = (1) → (2) → (3) → (4) → (5) → (6) → (9)和$path_4$ = (1) → (2) → (3) → (4) → (5) → (6) → (10)。经置信度评分筛选后，$path_1$因得分仅为2，低于预设阈值而被过滤。其余路径$path_2$、$path_3$和$path_4$共享共同节点(1)、(2)、(3)和(4)，且路径$path_3$和$path_4$进一步共享共同节点(5)和(6)。通过合并这些共同节点，重建攻击链结构如图7(c)所示。

$$\text{ConfidenceScore} = w_1 \text{LengthScore} + w_2 \text{NodeScore} \quad (9)$$

3　实验与分析

使用公开数据集和课题组通过模拟APT攻击采集的数据集对本文方法进行实验评估，旨在回答以下研究问题（research question, RQ）。

（1）RQ1：与其他先进技术相比，本文方法在攻击路径推理方面的效果如何？

（2）RQ2：本文方法的各关键模块对攻击路径推理效果的贡献如何？

（3）RQ3：在不同攻击案例中，本文方法的攻击路径推理表现如何？

（4）RQ4：参数设置如何影响本文方法的性能？

（5）RQ5：本文方法的运行效率如何？

3.1　实验设置

在一台Windows 10主机（CPU i5-10400F，内存16 GB）上部署Kellect[18]以采集内核日志数据，并在一个配备Intel Xeon Gold 5218 CPU、128 GB内存的Ubuntu 20.04.6服务器上运行本文方法。

3.1.1　APT模拟攻击数据集

课题组在部署的系统中模拟了5种包含ATT&CK多阶段攻击技战术的APT攻击，模拟攻击期间系统同时执行视频直播、办公软件、游戏及文件下载等常规活动，最终采集的内核日志数据包含超过1亿条事件[18]。所分析的5个攻击案例均代表一次完整的攻击，具体如下。

（1）案例1（Phishing Emails）：攻击者通过启用宏的钓鱼邮件发起攻击以实现初始访问。首先，在成功执行恶意代码后，攻击者通过在启动目录中部署VBScript脚本以建立持久化。随后，攻击者使用BloodHound进行环境侦察，并滥用cmstp实用程序进行权限提升。接下来的凭据窃取阶段使用Kerbrute工具，通过删除TeamViewer相关日志记录来规避防御检测。最终，被窃取的数据通过SMTP外传。

（2）案例2（Account Abuse）：首先，攻击者窃取现有凭据后，滥用WMI的Invoke-CimMethod方法创建隐藏计划任务。随后，将恶意脚本svchost.exe部署到System32目录以实现持久化，并利用卷影拷贝服务绕过注册表锁定，提取SAM/SYSTEM备份以进行密码破解。最终，攻击者通过加密Web服务外泄凭据。

（3）案例3（Content Injection）：首先，攻击者通过伪装成合法DNS、HTTP或SMB响应来投递恶意脚本，以此获得初始访问权限。随后，执行attrib命令修改系统文件属性以隐匿恶意负载。接着，使用动态目录服务接口（active directory ser-

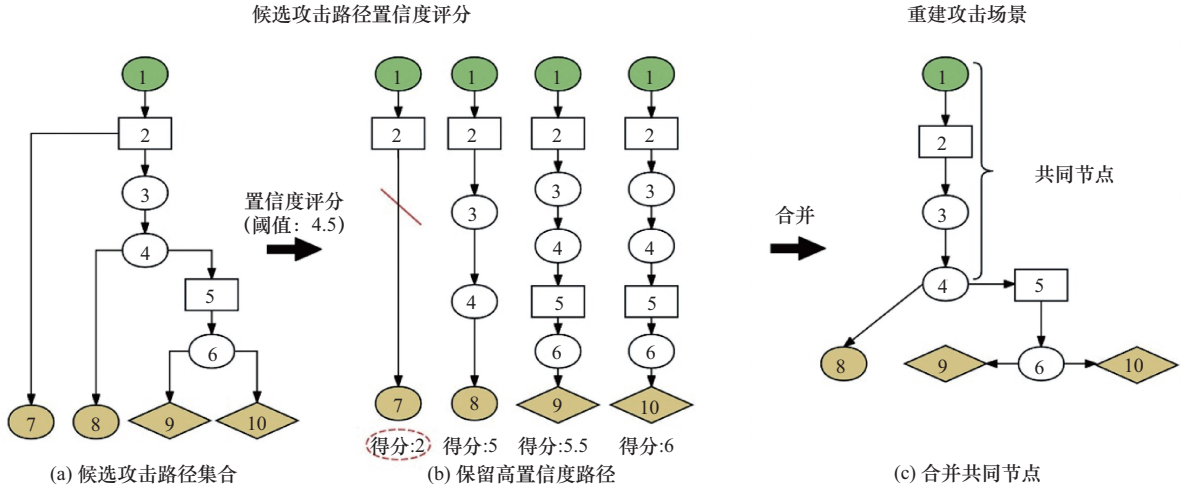


图7　攻击路径评分与合并示例

vice interfaces, ADSI）进行域内侦查，为横向移动收集信息，并最终通过HTTP协议外泄账户数据。

（4）案例4（Content Injection 2）：攻击者通过恶意链接投递一个木马化的可执行文件（如setup.exe），用户运行该文件会将DLL注入系统进程，并生成一个恶意启动脚本（startup.cmd）以实现持久化。在执行恶意代码后，攻击者使用PowerShell删除生成的脚本、日志和注册表项，以规避安全工具的检测。

（5）案例5（Service Suspension）：首先，攻击者部署带有固件级植入物的恶意USB设备以获取初始访问权限。随后，利用Windows管理工具（如SchTasks或WMI）创建计划任务并建立持久化，通过命令行禁用关键安全服务，最终部署高级加密标准（advanced encryption standard, AES）以实施系统破坏。

3.1.2 公开数据集

此外，在著名的公开数据集DARPA TC上评估本文方法的有效性。该数据集是威胁检测研究领域的成熟基准，包含由CADETS、THEIA和CLEARSCOPE这3种不同机制收集的数据。与真实的企业日志不同，DARPA TC数据集具有噪声干扰更低和恶意节点分布更稀疏的特点[19]。关于该数据集的具体测试案例，将在“案例研究”中详细阐述。

3.1.3 基准构建与评估指标

针对每个攻击案例，通过人工分析标注完整的攻击链，为攻击路径推理效果提供全面评估依据。具体而言，在采集的系统事件中执行节点级异常检测，保留异常节点并对图进行剪枝处理，仅保留关键依赖边，最终得到聚焦于攻击行为的精简异常子图。

表2统计了基于DARPA TC数据集生成的5类主机攻击[13]与5类自行模拟攻击案例的异常子图特征。其中，“#V”和“#E”分别表示原始溯源图中节点和依赖边的数量，“#CV”和“#CE”分别表示异常检测精简后保留的异常节点和边的数量。数据显示，原始溯源图中存在大量与真实攻击无关的节点与边。通过节点级异常检测与图精简，生成的异常子图的复杂度显著降低，有效提升了后续攻击路径推理的效率与清晰度。

本文采用精确率（Precision）、召回率（Recall）、F1分数（F1-Score）和漏报率（false negative rate, FNR）4种评估指标，$\text{Precision}=\frac{\text{TP}}{\text{TP}+\text{FP}}$，$\text{Recall}=\frac{\text{TP}}{\text{TP}+\text{FN}}$，$\text{F1-Score}=\frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$，$\text{FNR}=\frac{\text{FN}}{\text{TP}+\text{FN}}$，其中TP表示正确识别的攻击节点，FN表示被遗漏的攻击节点，FP表示被误报的良性节点。

表2 所有攻击生成的异常子图统计信息

数据集	攻击案例	#V	#E	#CV	#CE
模拟攻击	Phishing Emails	184 270	729 749	291	293
模拟攻击	Account Abuse	103 098	415 455	155	164
模拟攻击	Content Injection	160 904	603 197	216	219
模拟攻击	Content Injection 2	95 026	409 431	207	207
模拟攻击	Service Suspension	92 949	386 017	160	158
DARPA	Five Dir Case 1	8 572	25 084	327	530
DARPA	Five Dir Case 3	79 255	467 659	398	631
DARPA	Theia Case 1	303 704	1 107 028	2 287	2 609
DARPA	Theia Case 3	1 138 742	1 666 460	4 507	8 538
DARPA	Theia Case 5	53 779	400 123	338	458
平均值		222 029.9	621 020.3	888.6	1 380.7

3.2 对比实验

为验证本文方法在攻击路径推理方面的有效性，本节将其与前沿方法SPARSE[20]和DepImpact[17]进行实验对比。

（1）SPARSE：一种结合实时状态传播与路径级分析的混合方法。该方法基于轻量级语义规则，通过自动标签传播构建可疑语义图，并从中提取可疑路径以形成关键组件图。与本文方法深入挖掘攻击行为模式的研究思路不同，SPARSE更注重高效的证据筛选，即快速识别与兴趣节点关联的关键事件序列，而不解析攻击策略本身。

（2）DepImpact：一种基于多特征建模与影响力传播的图精简方法。该方法通过系统特征计算边权重，并沿依赖关系从兴趣节点向后进行影响力传播以识别关键路径。与本文方法致力于重建具有攻击技战术语义的攻击场景不同，DepImpact采用数据特征驱动的技术路线，即通过量化特征评分优先识别与兴趣节点相关的事件序列，其关注点在于系统级因果分析而非全局攻击意图的解读。

由于SPARSE与DepImpact均需以兴趣节点作为分析起点，本文统一采用3.2.1节检测到的异常节点作为兴趣节点。由表3可知，本文方法在各项评估指标上均显著优于基线方法，具体表现为攻击

表3　不同方法攻击路径推理

攻击案例	本文方法			SPARSE			DepImpact		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Phishing Emails	84.21%	100.00%	91.43%	28.57%	25.00%	26.67%	50.00%	80.00%	61.54%
Account Abuse	78.57%	91.67%	84.62%	40.00%	33.33%	36.36%	66.67%	80.00%	72.73%
Content Injection	75.00%	81.82%	78.26%	37.50%	27.27%	31.58%	20.00%	66.67%	30.77%
Content Injection 2	86.67%	76.47%	81.25%	38.89%	41.18%	40.00%	11.11%	50.00%	18.18%
Service Suspension	69.23%	90.00%	78.26%	37.50%	30.00%	33.33%	58.33%	70.00%	63.64%
Five Dir Case 1	37.50%	100.00%	54.55%	42.86%	100.00%	60.00%	33.33%	100.00%	50.00%
Five Dir Case 3	35.29%	100.00%	52.17%	27.27%	100.00%	42.85%	20.69%	100.00%	34.29%
Theia Case 1	25.81%	100.00%	41.03%	18.42%	87.50%	30.43%	12.31%	100.00%	21.92%
Theia Case 3	22.12%	89.29%	35.45%	33.87%	75.00%	46.67%	24.35%	100.00%	39.16%
Theia Case 5	30.77%	80.00%	44.45%	29.41%	100.00%	45.45%	27.78%	100.00%	43.48%
平均值	54.52%	90.93%	64.15%	33.43%	61.93%	39.33%	32.46%	84.67%	43.57%

节点还原率更高且引入的噪声更少。特别值得关注的是，本文方法的F1-Score达到64.15%，较SPARSE（39.33%）与DepImpact（43.57%）实现大幅提升，这充分证明了本文方法在高精度还原APT攻击链方面具有较好的鲁棒性。

现有基线方法的性能差距主要归因于以下两点。

首先，SPARSE基于语义与时序关系提取可疑路径，难以有效过滤与攻击节点相邻的良性节点，导致混入大量无关节点，从而降低了攻击路径推理的精确率。此外，对于多阶段复杂攻击链，仅依靠语义与时序关系进行攻击路径推理难以实现完整攻击链还原。如表3所示，SPARSE生成的攻击路径（总计204个节点）中误报节点达141个，致使其综合性能表现不佳（F1-Score=39.33%）。

其次，DepImpact在召回率（84.67%）方面表现出色，但精确率（32.46%）显著偏低，这表明该方法存在较高的误报倾向。原因在于其启发式策略通过选取前向与后向依赖图的交集子图作为输出。虽然这种设计能有效覆盖多阶段攻击，但会不可避免地引入大量无关节点，导致检测精度下降。

本文方法取得了64.15%的F1分数，其优势主要源于两个关键设计：（1）通过引入攻击技战术序列模式，系统性地过滤了不属于同一APT攻击的干扰节点，显著提升了检测精度（特别是同一个溯源图中包含多个攻击的情况）；（2）综合考量局部依赖边语义与全局路径语义，通过计算路径置信度有效区分不同攻击产生的攻击路径。

综上，针对RQ1，与现有先进方法DepImpact和SPARSE相比，本文方法通过引入攻击技战术序列模式来约束攻击路径的推理过程，在保持较高召回率的同时，显著提升了精确率。

3.3　消融实验

为评估本文方法各关键模块对攻击路径推理的贡献，本节将其与3种变体进行对比。

（1）w/o TTP：该变体移除了攻击技战术识别模块，通过随机分配攻击技战术类型至异常子图节点来模拟弱攻击语义识别场景。

（2）w/o PS：该变体移除了路径评分模块，不对路径推理获得的候选攻击路径进行置信度评估，而是直接将所有候选攻击路径合并作为最终重建的攻击链。

（3）w/o ATT-SPT：该变体不采用挖掘的攻击技战术序列模式树（ATT-SPT）指导攻击路径推理，而改用简单的攻击序列集合作为替代。

表4的实验结果展示了本文方法各核心机制的作用，其在F1-Score上分别以72.73%、76.85%和51.64%的显著优势超越所有变体，这些差距说明了3个问题。首先，节点级攻击技战术识别构成攻击识别的语义基础。当采用随机标注替代时，系统失去对攻击技战术的逻辑理解能力，导致路径推理性能下降72.73%。其次，路径置信度评估是把握攻击路径精确率的核心。未引入该机制的变体虽实现100%的召回率，却以10.72%的极低精确率为代价，说明单纯依赖语义特征与威胁权重的攻击路径推理会保留大量无关节点。再次，ATT-SPT为边评分提供了关键的语义指导。当改用简单的攻击序列

集合时，F1-Score 出现显著下降（下降 51.64%），这证实了基于攻击技战术知识的模式推理在识别复杂攻击路径关联性方面具有显著作用，其层次化结构不仅能捕捉直接攻击转换，更能识别跨战术层的多跳攻击依赖关系。

综上，针对 RQ2，本文方法的攻击路径推理效果由 3 个核心模块（攻击技战术识别、攻击技战术序列模式树和攻击路径评分）协同工作保证，缺少任何一个模块均会导致性能显著下降。

3.4　案例分析

鉴于 APT 攻击路径推理的复杂性，仅定量实验分析难以理解本文方法具有优势的原因和适用的场景。为此，同时进行定性实验分析，对多个实际案例进行深入剖析。之前的实验结果表明，本文方法在攻击链路还原过程中能显著降低误报与漏报。现有方法的推理误差主要源于攻击场景的固有复杂性：（1）存在多组具有复杂耦合关系的相互依赖的攻击链；（2）攻击链横跨多个攻击阶段。基线方法难以有效处理此类问题，因为它们仅依赖局部语义或上下文节点信息，本质上缺乏识别全局攻击意图模式的能力。典型案例如下：当遇到共享公共节点的交叉攻击路径时，基于静态拓扑特征的 SPARSE 和 DepImpact 会将其简单归类为单一攻击路径；本文能依据不同的攻击技战术序列模式区分不同的攻击路径，将看似交叉的攻击路径有效拆解为独立攻击路径。通过分析以下实验案例说明这一结论。

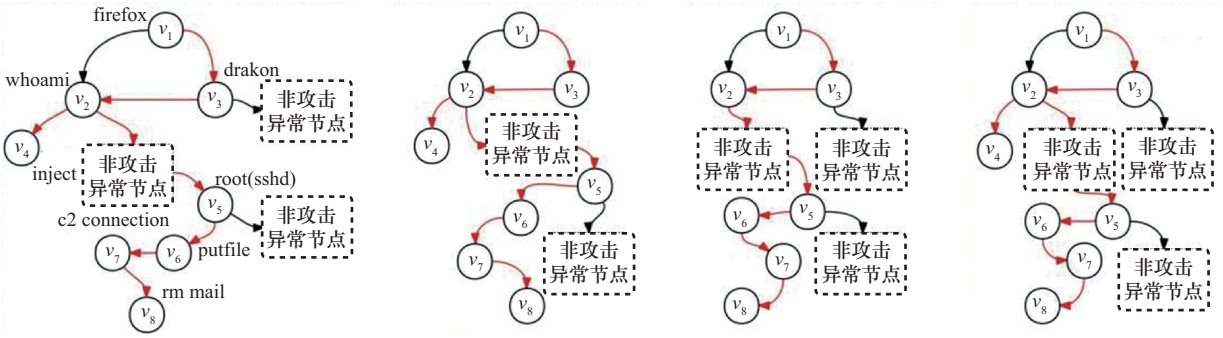
案例 1：如图 8 所示，该案例来自 DARPA TC 数据集中的“Theia Case 1”，记录了一次典型的客户端 APT 攻击。攻击者先利用 Firefox 浏览器漏洞植入后门程序，随后与远程命令控制（C2）服务器建立持久化反向 Shell 连接。

针对案例 1，本文方法、SPARSE 与 DepImpact 均展现出稳定的检测能力，各方法得到的攻击路径均实现了较高的攻击链覆盖率，能够完整捕捉到基本攻击序列。通过对案例的深入分析，发现该攻击链呈现两个关键特征：其一，攻击路径呈现线性单链依赖结构；其二，真实攻击节点稀疏分布在大量良性节点中。换言之，DARPA 数据集中的多数溯源图仅包含单一攻击，模型只需按时间顺序连接恶意节点并过滤良性节点即可。在此条件下，3 种方法均实现了完整的攻击链覆盖，性能表现相当。

案例 2：该案例来自课题组的模拟 APT 攻击“Phishing Emails”。该攻击先通过钓鱼邮件发起凭

表 4　　**本文方法及其变体对攻击路径推理的效果对比**

攻击案例	本文方法			w/o TTP			w/o PS			w/o ATT-SPT		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Phishing Emails	84.21%	100.00%	91.43%	42.86%	18.75%	26.09%	10.46%	100%	18.93%	42.86%	18.75%	26.09%
Account Abuse	78.57%	91.67%	84.62%	25.00%	22.22%	23.53%	7.79%	100%	14.46%	62.50%	55.56%	58.82%
Content Injection	75.00%	81.82%	78.26%	30.77%	36.36%	33.33%	9.02%	100%	16.54%	45.45%	50.00%	47.62%
Content Injection 2	86.67%	76.47%	81.25%	11.11%	5.88%	7.69%	8.29%	100%	15.32%	22.22%	11.76%	15.38%
Service Suspension	69.23%	90.00%	78.26%	28.57%	18.18%	22.22%	18.03%	100%	30.56%	46.15%	60.00%	52.17%
平均值	78.74%	87.99%	82.76%	27.66%	20.28%	22.57%	10.72%	100%	19.16%	43.84%	39.21%	40.02%



(a) Theia Case 1　(b) 本文方法　(c) SPARSE　(d) DepImpact

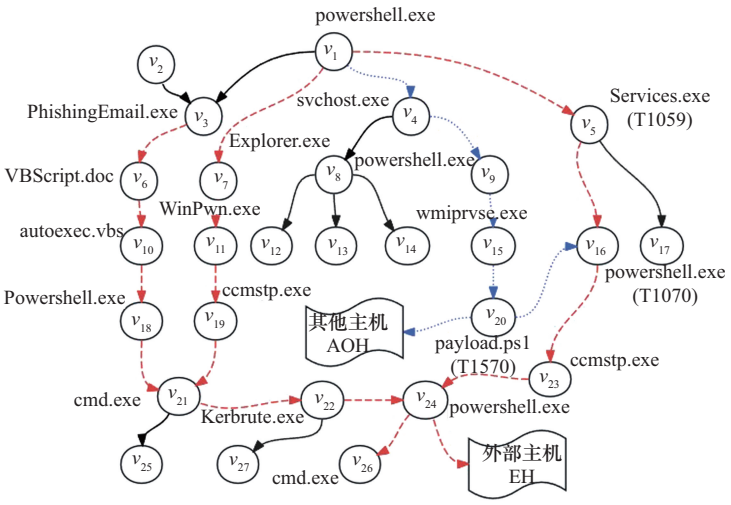
图 8　本文方法与基线方法对案例 1 重建的攻击链分析

据窃取，随后建立持久化访问、实施权限提升，并系统性地收集与外泄敏感数据。该场景包含一个进程伪装链，通过将恶意执行流嵌入合法系统进程序列中来增加取证分析难度。

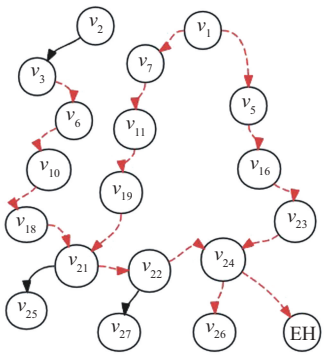
针对案例2，本文方法在分析耦合攻击链方面展现出显著优势，其精确率与召回率均显著优于SPARSE与DepImpact。在实际网络环境中，多组攻击可能同时发生，甚至共享部分节点（如通用进程）。如图9所示，Phishing Emails攻击案例构建的异常子图中存在两条攻击路径：虚线路径表示钓鱼攻击的真实攻击链，点线路径则表示另一条攻击行为伪装链。这两条攻击链在关键进程节点v_{16}（标记为T1070）处交汇，该节点被不同攻击链复用以实现不同的攻击目的。本文方法采用攻击技战术序列模式指导攻击路径推理，当回溯至v_{16}节点时检测到两个前置节点：v_5（标记为T1059）和v_{20}（标记为T1570）。基于ATT-SPT的关联分析，由于T1059→T1070的攻击技术转换概率显著高于T1570→T1070，攻击路径推理过程中优先选择v_5作为v_{16}的回溯节点而忽略v_{20}，从而有效避免了可能引发场景歧义的攻击链耦合问题。

相比之下，SPARSE主要关注语义关系，其攻击路径推理依赖于兴趣节点质量及其时序依赖关系。例如，当从外部主机节点向v_{24}进行反向推理时，由于$v_{23}\rightarrow v_{24}$在时序上优先于$v_{22}\rightarrow v_{24}$，左侧关键攻击链被忽略。然而，若能识别多个有效的兴趣节点，SPARSE仍可成功还原此攻击路径到完整攻击链路中，这表明该方法更适用于单链攻击场景。DepImpact通过兴趣节点与入口节点间的双向因果分析，能够实现对多攻击链的更全面覆盖。然而，当面对存在耦合关系的攻击链时，该方法难以有效区分不同攻击路径的独立攻击意图，限制了其在多链攻击场景下的有效性。

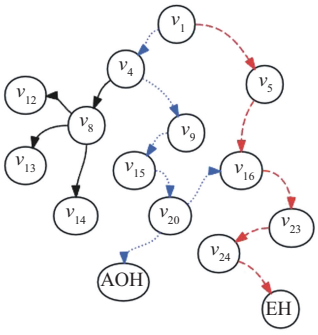
综上，针对RQ3，在存在攻击链耦合、攻击链



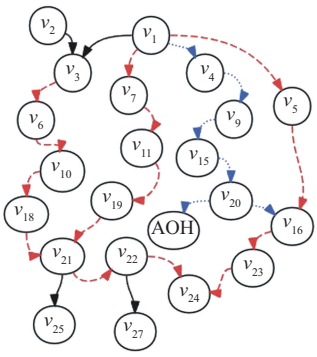
(a) Phishing Emails



(b) 本文方法



(c) SPARSE



(d) DepImpact

图9　本文方法与基线方法对案例2重建的攻击链分析

误报等干扰的复杂场景下，本文方法能够借助攻击技战术知识更有效地捕捉到真实的攻击路径。

3.5　调参实验

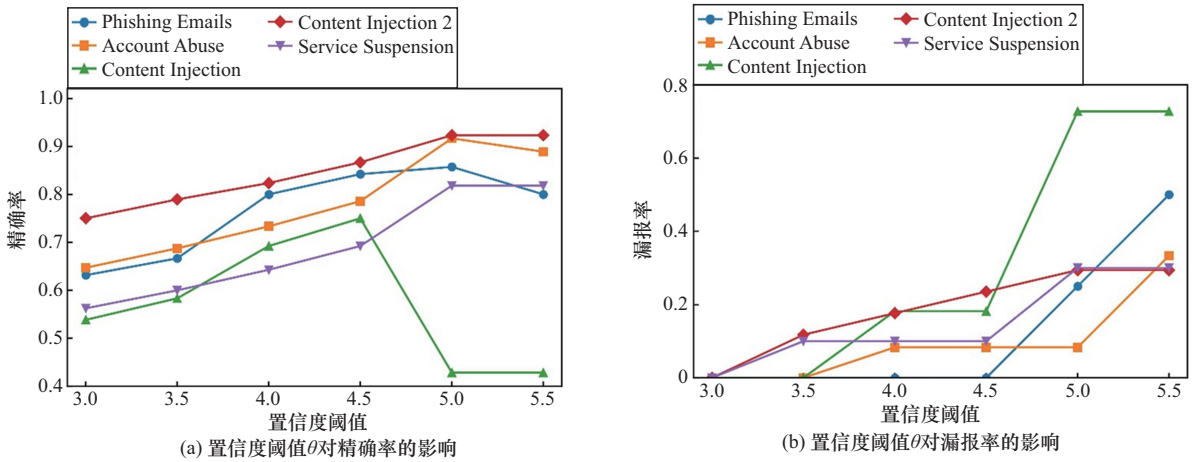
置信度阈值 θ（第 2.4 节）在平衡检测灵敏度与精确率方面起到了关键作用：较低阈值能保留更多攻击相关节点但同时也容易引入噪声，而较高阈值虽能获得更具威胁的精简路径却可能遗漏关键攻击节点。为评估此权衡关系，可以通过调节该参数来观察精确率与漏报率的变化趋势。

如图 10 所示，当置信度阈值 θ 设置较低（3.0~3.5）时，重建的攻击链中会保留更多候选推理路径，虽能实现接近零的漏报率，但精确率始终表现欠佳。随着置信度阈值 θ 逐步提升，精确率逐渐攀升至峰值，但与此同时漏报率急剧上升。这种现象源于对候选推理路径的过度剪枝，导致重建的攻击链过度简化而遗漏真实攻击节点。

综上，针对 RQ4，关键参数置信度阈值 θ 用于平衡攻击路径推理的完整性和精确性，通过调参实验最终设置置信度阈值 $\theta = 4.5$。

3.6　计算效率实验

为评估本文方法的计算效率，系统测算了其全流程关键模块的执行耗时，如表 5 所示，其中完成一次完整攻击路径推理（涵盖异常子图挖掘、威胁权重分配与攻击链重建）平均耗时 152.07 s。与 SPARSE 和 DepImpact 的对比分析表明：（1）本文方法基于 ATT-SPT 的威胁权重分配环节（2.35 s）虽在耗时上略高于 DepImpact 的依赖传播算法（0.81 s），但其通过攻击技战术转换概率框架指导威胁权重计算，以时间代价换取了 67.9% 的攻击路径推理精确率提升；（2）在攻击链重建阶段，本文方法凭借路径置信度评分机制（通过置信度阈值过滤无关攻击行为），仅需 0.002 2 s 即可完成，效率比 SPARSE（0.056 s）快近 25 倍，比 DepImpact（1.07 s）快近 486 倍。尽管本文方法在异常子图挖掘阶段耗时略长，但其基于攻击技战术序列模式的威胁评估模型最终实现了卓越的攻击链路还原度，推理准确率达 54.52%，召回率高达 90.93%。



(a) 置信度阈值θ对精确率的影响　　(b) 置信度阈值θ对漏报率的影响

图 10　调参实验结果

表 5　本文方法与基线方法的平均耗时对比

攻击案例	异常子图挖掘/s	威胁权重分配		攻击链重建		
		本文方法/s	DepImpact/s	本文方法/s	SPARSE/s	DepImpact/s
Phishing Emails	234.82	2.60	0.82	0.003	0.06	1.35
Account Abuse	137.45	2.43	0.96	0.002	0.05	1.03
Content Injection	134.61	1.77	0.44	0.001	0.05	0.86
Content Injection 2	123.97	2.71	0.94	0.003	0.06	1.12
Service Suspension	117.74	2.23	0.91	0.002	0.06	0.99
平均值	149.72	2.35	0.81	0.002 2	0.056	1.07

综上，针对RQ5，本文方法在计算开销上是可接受的，其主要耗时集中在前期异常子图挖掘阶段，而在攻击路径推理阶段具有较高效率。

3.7　讨论

（1）实际应用。攻击溯源技术的核心目标在于还原能够完整覆盖恶意活动序列与关系的攻击链路。作为一个综合性的攻击调查框架，本文方法深度融合了多种先进技术，显著提升了对复杂多阶段攻击链的还原能力。通过将节点级异常检测与攻击技战术识别机制相结合[15]，本文方法能有效滤除大量与攻击无关的节点和边，并依据ATT&CK框架为异常节点标注对应的战术或技术，这些模块协同工作，使其能够沿潜在攻击路径实现快速、高效的推理，从而在整个攻击阶段内精准揭示高优先级的攻击行为。通过聚焦攻击场景图中最关键的要素，本文方法为威胁缓解与主动防御策略的制定提供了关键的决策依据。

（2）泛化能力。APT的核心特征在于其多阶段性，攻击者通过协调运用多种攻击战术与技术来渗透目标系统。这些战术与技术之间存在一定的内在关联性。本文方法通过挖掘历史攻击行为特征与最新对抗策略，有效提取战术与技术间关联关系，为构建ATT-SPT奠定基础。该方法还具备持续演进能力，通过吸纳新兴威胁手段产生的新型攻击模式，不断迭代优化序列模式库，从而加强对未知攻击场景的适应能力。

（3）局限性。尽管本文方法通过引入外部攻击技战术知识在攻击路径推理方面取得重要进展，但当前框架仍存在以下局限。首先，ATT&CK本身是持续演进的知识框架，不同版本对同一攻击行为的技术编号、战术归属及语义边界可能存在调整。同时，不同检测引擎基于各自规则库对同一底层行为给出的技术编号标注也可能不一致。本文在模拟数据集与DARPA数据集的实验中，默认攻击技战术标注处于统一知识空间，但在面向更复杂真实环境时，仍需进一步解决跨ATT&CK版本、跨检测引擎以及跨数据源之间的语义对齐问题，否则将影响ATT-SPT构建与路径评分的一致性。其次，本文当前利用技战术序列模式刻画攻击阶段演化规律，而对于威胁情报中更细粒度的工具、组织及其攻击者偏好等信息的利用仍然有限。事实上，不同攻击组织对特定工具链的依赖、不同工具之间的协同关系，以及部分技术在真实对抗中的稀缺性与专属性，均可为攻击路径推理提供更强的先验约束。因此，未来可进一步在ATT-SPT的基础上引入面向技术分类分级、工具依赖和组织偏好的差异化权重机制，以提升在复杂攻击应用场景下推理的准确性。最后，本文方法性能仍受底层异常检测模块性能的制约。若关键攻击节点在异常检测阶段被遗漏，则攻击链可能出现断裂。当前ATT-SPT主要用于边威胁评分与候选路径推理，尚未被显式用于缺失节点恢复。但从方法机制来看，其蕴含的攻击阶段演化模式可作为路径补全的先验知识：当两个已观测节点之间出现不连续的技战术转换时，可利用ATT-SPT推断可能缺失的中间技术或战术阶段，并结合时序约束、依赖关系可达性、节点异常分数和实体类型信息，在原始溯源图中检索合适的节点，从而增强攻击路径推理的鲁棒性。值得说明的是，本文通过多层次分析框架将离散异常节点与已知攻击技战术序列模式进行关联，有效缓解了异常检测遗漏与多链耦合对路径推理的影响，这已在3.2节对比实验、3.3节消融实验和3.4节案例分析中得到体现。

4　相关工作

基于溯源技术的攻击调查方法是还原APT攻击链路的关键手段。在图优化方面，NoDoze[21]通过威胁警报的自动分类，在保留关键攻击路径的同时显著压缩图规模，有效缓解安全分析师警报疲劳问题；PrioTracker[22]则采用基于节点关联影响程度的动态优先级调度策略，通过优先分析高风险路径减少人工核查边的数量，但该策略在复杂软件环境（如浏览器）中易对高连通度节点产生误判。针对分布式环境，跨主机溯源追踪通过关联多源审计日志（如系统调用、流量信息）实现APT横向移动链的完整重现，典型代表包括Milajerdi等[5]提出的信息流标记技术，以及Alsaheel等[23]提出的跨主机因果推理方法，有效解决了攻击路径碎片化问题。威胁情报增强型方案近年来备受关注：文献[24-26]证实通过主动威胁狩猎查询与溯源图匹配可实现对隐匿攻击的实时检测；Milajerdi等[4]与Zhao等[27]则利用结构化威胁情报进行语义节点标注[28]，通过与ATT&CK框架对齐，提升攻击意图识别精度。但现有方法仍存

在固有局限：基于兴趣节点为起点的反向因果分析易引发依赖爆炸[29]，传统剪枝启发式规则（如时序/数据流特征）因上下文的曲解会导致合法攻击边被错误剔除[9]。本文方法通过将细粒度溯源分析与外部威胁情报相结合，实现了更精准的攻击调查。该方法还具有良好的兼容性，在剪枝阶段可采用基于模板匹配的图压缩技术合并相似节点[30]，也可以结合日志垃圾回收机制[31]消除低价值日志事件，有效降低噪声干扰。

异常检测技术历经长期演进，早期系统主要依赖阈值驱动的统计启发式规则与特征匹配技术[32-33]。基于机器学习与深度学习的检测方法则显著减少了人工干预：监督学习与集成方法利用标注数据进行模型训练，但依赖高精度的标注数据[34-35]；无监督学习技术虽能有效识别日志序列中的统计异常，但其忽略语义特性导致难以解析事件间的关联含义[4,36-37]。图神经网络通过溯源图分析提升检测精度，但同时带来巨大计算开销[3,38]。APTSHIELD[39]创新性地构建了语义丰富的攻击链建模框架，将多阶段攻击语义抽象为可解释的语义转移模式以提升威胁归因能力。然而，现有方法仍主要聚焦节点级威胁检测，对复杂攻击场景中跨阶段威胁因果关系的探索尚显不足。本文方法创新性地融合节点级检测技术[15,40-43]与威胁情报增强机制，在有效控制误报的同时，实现了对多阶段攻击链的精确还原。

5 结束语

本文提出了一个融合攻击技战术进行攻击路径推理的方法，该方法通过因果分析从系统日志生成溯源图，在此基础上执行节点级异常检测与攻击技战术识别，保留异常节点及其关键依赖边，并为节点标注其攻击技战术类型。进一步，通过挖掘ATT-SPT及节点邻近交互关系对异常子图的依赖边进行威胁权重评分，在路径推理中根据威胁权重游走以获得候选攻击路径。该方法通过路径的多维特征评估路径置信度，保留并合并高置信度分数路径，最终还原攻击链得到重建的攻击链。在多个攻击案例上的评估表明，本文方法通过冗余子图剪枝、攻击路径推理与基于上下文语义的路径筛选流程，在攻击路径推理的准确率与完整度间取得了最佳平衡。

参考文献：

[1] Chen J G, Su C H, Yeh K H, et al. Special issue on advanced persistent threat[J]. Future Generation Computer Systems, 2018, 79: 243-246.

[2] Hossain M N, Milajerdi S M, Wang J A, et al. SLEUTH: real-time attack scenario reconstruction from COTS audit data[C]//USENIX Security Symposium. Berkeley: USENIX Association, 2017: 487-504.

[3] Han X Y, Pasquier T, Bates A, et al. Unicorn: runtime provenance-based detector for advanced persistent threats[C]//Proceedings 2020 Network and Distributed System Security Symposium. Virginia: The Internet Society, 2020: 23-26.

[4] Milajerdi S M, Eshete B, Gjomemo R, et al. POIROT: aligning attack behavior with kernel audit records for cyber threat hunting[C]//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2019: 1795-1812.

[5] Milajerdi S M, Gjomemo R, Eshete B, et al. HOLMES: real-time APT detection through correlation of suspicious information flows[C]//Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2019: 1137-1152.

[6] Jiang B X, Bilot T, Madhoun N E, et al. ORTHRUS: achieving high quality of attribution in provenance-based intrusion detection systems[C]//USENIX Security Symposium. Berkeley: USENIX Association, 2025: 7173-7192.

[7] Cheng X Y, Liu Z, Tang M, et al. Anomaly detection based on improved isolated forest[C]//Proceedings of the 2023 IEEE 3rd International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA). Piscataway: IEEE Press, 2023: 971-975.

[8] Wei R Z, Cai L J, Yu A M, et al. DeepHunter: a graph neural network based approach for robust cyber threat hunting[PP]. V1. (2021-04-20)[2026-02-09]. arXiv: arXiv.2104.09806.

[9] Hassan W U, Bates A, Marino D. Tactical provenance analysis for endpoint detection and response systems[C]//Proceedings of the 2020 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2020: 1172-1189.

[10] Al-Shaer R, Spring J M, Christou E. Learning the associations of MITRE ATT&CK adversarial techniques[PP]. V2. (2020-05-12)[2026-02-09]. arXiv: arXiv.2005.01654.

[11] The DFIR Report. Continuing the bazar ransomware story[R]. 2025-07-05.

[12] Venere G, Neal C. AVOSlocker's new arsenal, blog post[R]. 2022-06-21.

[13] DARPA. Transparent computing, defense advanced research projects agency[R]. 2025-10-29.

[14] Patil R, Muneeswaran S, Sachidananda V, et al. E-Audit: distinguishing and investigating suspicious events for APTs attack detection[J]. Journal of Systems Architecture, 2023, 144: 102988.

[15] Lyu M Q, Gao H Z, Qiu X B, et al. TREC: APT tactic/technique recognition via few-shot provenance subgraph learning[C]//Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2024: 139-152.

[16] Mohammadi S, Kermabon-Bobinnec H, Tabiban A, et al. Connecting the extra dots (contexts): correlating external information about point of interest for attack investigation[C]//Proceedings of the 2025 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2025: 130-148.

[17] Fang P C, Gao P, Liu C L, et al. Back-propagating system dependency impact for attack investigation[C]//Preceedings of the 31th USENIX Security Symposium. Berkeley: USENIX Association, 2022: 2461-2478.

[18] Chen T, Song Q, Qiu X, et al. Kellect: a kernel-based efficient and lossless event log collector for Windows security[J]. Computers & Security, 2025, 150: 104203.

[19] Sharafaldin I, Lashkari A H, Ghorbani A. Toward generating a new intrusion detection dataset and intrusion traffic characterization[C]//Proceedings of the 4th International Conference on Information Systems Security and Privacy. Piscataway: IEEE Press, 2018: 108-116.

[20] Ying J, Zhu T, Cheng W, et al. SPARSE: semantic tracking and path analysis for attack investigation in large-scale provenance graphs[J]. IEEE Transactions on Dependable and Secure Computing, 2026, 23(2): 1-15.

[21] Hassan W U, Guo S J, Li D, et al. NoDoze: combatting threat alert fatigue with automated provenance triage[C]//Proceedings of the NDSS Symposium 2019. Piscataway: IEEE Press, 2019: 1-15.

[22] Liu Y S, Zhang M, Li D, et al. Towards a timely causality analysis for enterprise security [C]//Proceedings of the 25th Annual Network and Distributed System Security Symposium (NDSS 2018). Piscataway: IEEE Press, 2018: 1-15.

[23] Alsaheel A, Nan Y H, Ma S Q, et al. ATLAS: a sequence-based learning approach for attack investigation[C]//Preceedings of the 30th USENIX Security Symposium. Berkeley: USENIX Association, 2021: 3005-3022.

[24] Pei K X, Gu Z S, Saltaformaggio B, et al. HERCULE: attack story reconstruction via community discovery on correlated log graph[C]//Proceedings of the 32nd Annual Conference on Computer Security Applications. New York: ACM Press, 2016: 583-595.

[25] Schwartz Y, Benshimol L, Mimran D, et al. LLMCloudHunter: harnessing LLMs for automated extraction of detection rules from cloud-based CTI[PP]. V1. (2024-07-06) [2026-02-09]. arXiv: arXiv. 2407.05194.

[26] Shah S, Parast F K. AI-driven cyber threat intelligence automation[PP]. V1. (2024-10-26)[2026-02-09]. arXiv: arXiv.2410.20287.

[27] Zhao J, Yan Q B, Liu X D, et al. Cyber threat intelligence modeling based on heterogeneous graph convolutional network[C]//Preceedings of the 23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020). Berkeley: USENIX Association, 2020: 241-256.

[28] Li J W, Zhang R, Liu J Y, et al. LogKernel a threat hunting approach based on behaviour provenance graph and graph kernel clustering[PP]. V1. (2022-08-18)[2026-02-09]. arXiv: arXiv.2208.08820.

[29] Xu Z, Wu Z Y, Li Z C, et al. High fidelity data reduction for big data security dependency analyses[C]//Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2016: 504-516.

[30] Tang Y T, Li D, Li Z C, et al. NodeMerge: template based efficient data reduction for big-data causality analysis[C]//Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2018: 1324-1337.

[31] Lee K H, Zhang X Y, Xu D Y. LogGC: garbage collecting audit log[C]//Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security. New York: ACM Press, 2013: 1005-1016.

[32] Manzoor E, Milajerdi S M, Akoglu L. Fast memory-efficient anomaly detection in streaming heterogeneous graphs[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2016: 1035-1044.

[33] Wang Q, Hassan W U, Li D, et al. You are what you do: Hunting stealthy malware via data provenance analysis [C]//Proceedings of the NDSS Symposium 2020. Piscataway: IEEE Press, 2020: 1-17.

[34] Gao P, Xiao X S, Li Z C, et al. AIQL: enabling efficient attack investigation from system monitoring data[C]//Proceedings of the 2018 USENIX Annual Technical Conference. Berkeley: USENIX Association, 2018: 113-126.

[35] Tang B H, Yang J L, Li X, et al. APT detector: detect and identify APT malware based on deep learning framework[C]//Proceedings of the 2023 9th International Conference on Computing and Artificial Intelligence. New York: ACM Press, 2023: 576-583.

[36] Du M, Li F F, Zheng G N, et al. DeepLog: anomaly detection and diagnosis from system logs through deep learning[C]//Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2017: 1285-1298.

[37] Mirsky Y, Doitshman T, Elovici Y, et al. Kitsune: an ensemble of autoencoders for online network intrusion detection[PP]. V2. (2018-05-27) [2026-02-09]. arXiv: arXiv.1802.09089.

[38] Yang F, Xu J C, Xiong C L, et al. PROGRAPHER: an anomaly detection system based on provenance graph embedding[C]//Proceedings of the 32nd USENIX Security Symposium. Berkeley: USENIX Association, 2023: 4355-4372.

[39] 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.

[40] 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.

[41] Goyal A, Wang G, Bates A. R-CAID: embedding root cause analysis within provenance-based intrusion detection[C]//Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2024: 3515-3532.

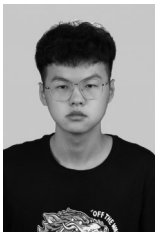
[42] Cheng Z J, Lv Q J, Liang J Y, et al. Kairos: practical intrusion detection and investigation using whole-system provenance[C]//Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP). Piscat-

away: IEEE Press, 2024: 3533-3551.

[43] Rehman M U, Ahmadi H, Hassan W U. Flash: a comprehensive approach to intrusion detection via provenance graph representation learning[C]//Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2024: 3552-3570.

[作者简介]

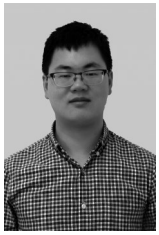
吕明琪（1982– ），男，浙江工业大学教授、博士生导师，主要研究方向为网络安全、数据挖掘。



盛起（2000– ），男，浙江工业大学硕士生，主要研究方向为网络安全、APT攻击溯源。

陈铁明（1978– ），男，浙江工业大学教授、博士生导师，主要研究方向为网络安全、人工智能。

朱添田（1992– ），男，浙江工业大学副教授，主要研究方向为网络安全、自然语言处理。



王飞（1989– ），男，中国石油大学（华东）博士生，主要研究方向为工业互联网、人工智能。