

文章编号:1671-5896(2026)04-0765-11

基于蚁群算法的通信网络节点攻击漏洞安全检测方法

李 强¹, 黄优哲¹, 魏 柳¹, 陈文琅²

(1. 南方电网电动汽车服务有限公司 平台部, 广东 深圳 518110;
2. 北京启明星辰信息安全技术有限公司 广东分公司, 广州 510000)

摘要: 针对在通信网络漏洞检测中, 因忽略簇结构信息使其精准度不足的问题, 提出一种基于蚁群算法的通信网络节点攻击漏洞安全检测方法。首先运用滑动时间窗口分割与时空互相关分析提取低维可解释节点特征, 增强异常模式区分度; 然后借助卷积神经网络对降维特征进行卷积、池化及展平操作, 并结合 Softmax 分类器判定异常节点概率, 缩小检测范围; 最后基于被动分簇算法划分的簇结构, 采用蚁群算法, 结合簇内节点关系与通信模式, 通过信息素浓度追踪漏洞位置。实验结果表明, 该方法能精准定位通信网络攻击漏洞, 可全面、准确评估网络安全状况, 为通信网络漏洞检测提供了有效解决方案。

关键词: 通信网络; 异常节点检测; 漏洞检测; 蚁群算法; 安全漏洞弧

中图分类号: TP393 **文献标志码:** A

Security Detection Method of Node Attack for Communication Network Based on Ant Colony Algorithm

LI Qiang¹, HUANG Youzhe¹, WEI Liu¹, CHEN Wenlang²

(1. Platform Department, China Southern Power Grid Electric Vehicle Service Company Limited, Shenzhen 518110, China;
2. Guangdong Branch, Beijing Venustech Cybervision Company Limited, Guangzhou 510000, China)

Abstract: In the detection of communication network vulnerabilities, the accuracy is insufficient due to the lack of cluster structure information. Therefore, a security detection method for communication network node attack vulnerabilities based on ant colony algorithm is studied. Firstly, sliding time window segmentation and spatiotemporal cross-correlation analysis are used to extract low dimensional interpretable node features and enhance anomaly pattern discrimination. Then, convolutional neural networks are used to perform convolution, pooling, and flattening operations on the reduced dimensional features, combined with a Softmax classifier to determine the probability of abnormal nodes and narrow down the detection range. Finally, based on the passive clustering algorithm, the cluster structure is divided using ant colony algorithm, combined with the relationships and communication modes of nodes within the cluster, to track the location of vulnerabilities through pheromone concentration. The experimental results show that this method can accurately locate communication network attack vulnerabilities. Therefore, this method can comprehensively and accurately evaluate the network security situation, providing an effective solution for communication network vulnerability detection.

Key words: communication network; abnormal node detection; vulnerability detection; ant colony; security vulnerability arc

收稿日期: 2025-07-08

基金项目: 南方电网电动汽车服务有限公司网络安全监督运营辅助服务基金资助项目(CG3800022001914650)

作者简介: 李强(1990—), 男, 江西南丰人, 南方电网电动汽车服务有限公司工程师, 主要从事信息网络研究, (Tel) 86-15626436939 (E-mail) liqiang202504@126.com。

0 引言

通信网络节点的主要任务包括数据运输、处理和储存等,并且其是网络的重要组成部分之一^[1-2]。通信网络中攻击者通过节点入侵和破坏网络对漏洞进行攻击,若节点遭到破坏,不仅会影响服务的连续性和数据的隐私性,而且严重威胁整个网络运行的安全和稳定。因此,网络空间安全和稳定运行的关键是提升通信网络节点的安全检测能力。

在现有的通信网络漏洞检测研究中,各类方法从不同角度对节点漏洞进行了分析与识别。对基于程序依赖分析的漏洞检测方法,杨宏宇等^[3]通过构建控制与数据依赖链,并以此为基础构建漏洞检测神经网络模型,实现了对漏洞特征依赖图的学习。但该方法在构建依赖链时未充分考虑代码的簇结构特征,导致依赖关系提取不完整,进而影响漏洞检测的准确性。对基于图表示学习的漏洞检测方法,Yang等^[4]提出采用图结构表示网络代码,并建立门控图神经网络以捕捉节点间的消息传递模式。但该方法未针对网络簇结构中的局部通信模式进行专门优化,致使簇内小范围异常行为在全局特征学习中易被忽略,降低了检测的敏感度。对基于代码切片与图神经网络的混合方法,胡雨涛等^[5]提出对C/C++源代码进行程序切片处理,提取切片信息后输入图神经网络模型以获取漏洞特征。但该方法侧重于代码静态特征的提取,未结合网络运行时的动态行为信息。此外,由于未对不同角色节点赋予差异化权重,导致关键节点漏洞可能被忽略,或正常通信被误判。对基于多神经网络集成的检测框架方法,Chen等^[6]结合多个神经网络构建基础检测器框架,有效整合了不同类型的漏洞特征。然而,在多节点协作的簇结构中,局部异常行为可能因全局图卷积操作而被平滑,同时稀疏通信模式易被误判为异常,造成检测准确率下降。

虽然上述方法在通信网络漏洞检测方面取得了一定进展,但由于普遍忽视网络簇结构特性,存在漏洞特征提取不完整、局部异常响应迟缓、关键节点识别能力不足等问题,限制了检测精度的提升。为此,笔者提出一种通信网络节点攻击漏洞安全检测方法,突破了传统方法的簇结构盲区。首先,通过滑动窗口互相关系数与时间模糊性优化特征提取,构建轻量化卷积神经网络(CNN: Convolutional Neural Network)模型,利用局部特征降维加速异常分类,解决全局特征学习导致的局部异常丢失问题;其次,融合被动分簇与蚁群算法,结合虚拟移动定位技术精确计算节点间安全漏洞弧段,通过信息素浓度追踪实现漏洞快速定位,克服传统方法对簇内通信模式适配不足的缺陷。实验表明,该方法有效缩短了检测周期,提高了漏洞发现精度,为通信网络安全提供了更精准、高效的检测方案。

1 通信网络异常节点检测

1.1 通信网络节点特征提取

在通信网络节点特征提取过程中若直接处理原始高维数据,会忽略时空关联性与噪声冗余对检测精度的影响,尤其在面对伪装成正常流量的攻击时,难以有效揭示异常模式。为此,笔者提出基于滑动窗口互相关系数与时间模糊性的特征提取方法。通过分割滑动时间窗口并计算互相关系数,结合归一化处理消除节点距离变化对特征准确性的干扰,解决因空间互相关系数非对称性导致的特征丢失问题;引入时间模糊性映射对历史子时间序列重新规划,突出异常行为的时间维度差异,将高维原始数据转化为低维、可解释的特征向量。由此降低计算复杂度,并提升异常与正常行为的可分性,从而为后续检测模型提供精准的输入。

若多个传感器能共同探测到同一事件,则可依据该事件源的位置,确定这些传感器之间的相对距离。基于此,分割原始滑动时间窗口,获得若干个子序列。通过

$$V(v) = \text{Norm}(G_{\Xi}, G_{\Xi+1}) \quad (1)$$

确定监测互相关系数 $V(v)$ 。其中 $G_{\Xi}, G_{\Xi+1}$ 分别表示滑动前、后的子时间窗口; $\text{Norm}()$ 表示归一化计算。由于空间互相关系数是非对称的,当其在节点距离变化区间内,观测序列能被放置在固定的时间窗口中,从而提高通信网络节点特征提取的准确率。

相关系数在固定的监测时间窗口通过滑动完成遍历工作,获取监测数据。依据相对应的监测系数值

获取互相关系数的极值,识别重要的节点特征^[7]。选取监测节点,记录 and 解析其空间相关特征向量 $\mathbf{R}(V)$:

$$\mathbf{R}(V) = \text{Norm}(V_L, D_L), \quad (2)$$

其中 V_L 表示目前正常时间片段的时间维度; D_L 表示异常时间片段的时间维度。时间模糊性是以时间维度为根据,重新规划历史时间片段,获取节点特征 ΔA :

$$\Delta A = \frac{\cos \vartheta V_L}{X}, \quad (3)$$

其中 ϑ 表示模糊映射值; X 表示节点的历史子时间序列数据。

通过上述时空分析与模糊映射的方式,将高维原始数据转化为低维、可解释的特征向量,突出异常与正常行为的差异,为后续基于 CNN 的异常节点检测提供可分性更强的输入。

1.2 基于卷积神经网络的异常节点检测

由于卷积神经网络多采用全局特征学习,易忽略局部异常特征,从而导致对伪装攻击的识别率较低。为此,笔者提出轻量化 CNN 模型与局部特征降维策略。通过卷积层提取节点特征的局部模式^[8],利用线性整流函数增强非线性表达能力,避免梯度消失;池化层进一步降低特征维度,保留关键局部信息,解决全局特征学习导致的局部异常丢失问题^[9];全连接层将特征投影至低维子空间,缩短网络训练时间,在保持高精度的同时,显著减少模型参数规模,适用于资源受限的通信网络环境,有效保障 CNN 模型在异常检测中的效率与精度平衡。

将降维后的节点特征 x_i 输入网络卷积层中,对其进行处理获取局部特征 y_j^l :

$$y_j^l = g\left(\sum_{i=1}^j e_{i,j} x_i^{l-1} + n_j^l\right), \quad j = 1, 2, \dots, N, \quad (4)$$

其中 $e_{i,j}$ 表示卷积核对应的权重值; $j_{w,b}(x) = g(r^T x + b)$ 表示第 $l-1$ 层的第 i 个节点特征示意图; n_j^l 表示 l 层的第 j 个节点特征图的偏置参数; N 表示第 l 层的节点特征示意图的数量; $g(\cdot)$ 表示线性整流函数。线性整流函数通过引入非线性激活机制显著增强神经网络的表达能力,同时有效缓解梯度消失问题。其核心在于分段线性设计:当输入为正时,输出与输入保持线性关系(梯度为 1),确保前向传播中正区信号无损传递;当输入为负时,输出强制归零(梯度为 0),形成稀疏激活。这种特性使深层网络中梯度能逐层有效回传,避免传统 Sigmoid/Tanh 函数因饱和区梯度趋近于零导致的链式求导衰减。此外, $g(\cdot)$ 的计算复杂度低,加速了训练收敛。则有:

$$g(x) = \max(0, x). \quad (5)$$

将卷积层输出的局部特征 y_j^l 输入池化层中,对其进行池化处理,降低局部特征的维数。池化层的输出为

$$y_k^z = g(\text{down}(y_k^{z-1})), \quad (6)$$

其中 y_k^{z-1} 表示第 $z-1$ 层的局部特征表示图; $\text{down}(\cdot)$ 表示采样函数。

利用全连接层对上述局部特征 y_k^z 进行展平操作,获取有利于异常节点识别的特征 $j_{w,b}(y)$:

$$j_{w,b}(y_k^{z-1}) = g(r^T y_k^{z-1} + b), \quad (7)$$

其中 $j_{w,b}(y)$ 、 y_k^{z-1} 表示神经元的输出值和降维处理后的局部特征; r 表示神经元中所占的比重; b 表示偏置。

将上述获取的特征输入网络输出层中,利用 softmax 分类器对其进行判别,确定节点属于异常节点的概率:

$$p = \frac{e^{j_{w,b}(y_k^{z-1})}}{\sum_{z=1}^M e^{j_{w,b}(y_k^{z-1})}}. \quad (8)$$

根据式(8)可确定通信网络中的异常节点。对上述异常节点进行漏洞检测,可有效缩小网络漏洞检测的范围,提高漏洞检测精度。

1.3 基于蚁群优化与被动分簇的漏洞安全检测方法

将被动分簇或蚁群算法单独应用于漏洞检测时,存在如下 2 个缺陷:一是被动分簇算法对簇内通信

模式适配不足,难以精准定位失效节点;二是蚁群算法在全局搜索中易陷入局部最优,导致漏洞定位效率较低。在由被动分簇形成的无线传感器网络中,安全漏洞弧段是指一个簇内相邻两个节点的监测范围(感知半径)未能完全覆盖的间隙区域。该区域由于未被任何节点的感知范围所覆盖,构成了网络中的一个局部盲区,是潜在的安全漏洞。为此,笔者融合被动分簇与蚁群算法,提出基于簇结构信息的漏洞安全检测方法。首先,通过被动分簇算法划分网络节点,结合虚拟移动定位技术计算节点间安全漏洞弧段,明确簇内节点相对位置关系^[10],克服传统方法对簇结构信息利用不足的问题;其次,蚁群算法在簇内进行信息素浓度追踪,以标签值、距离和信息素浓度为转移概率依据,优先遍历高风险节点,形成最小周期环对应漏洞弧段,实现漏洞快速定位。由此,充分利用了簇内通信模式的局部性特征,显著缩短了检测周期,提高了漏洞发现的精度与效率。

在被动分簇算法中,若节点在通信网络的簇内失效,该节点将失去处理数据的能力,此时需要以中间节点为桥梁展开信息传输。设 $W(x)$ 表示通信网络节点出现安全漏洞的概率,如下:

$$W(x) = \iota D(x), \quad (9)$$

其中 ι 表示比例系数; $D(x)$ 表示节点 x 对应的映射区域内存在的实际簇首数量。一个节点的漏洞风险值,与其周围所有关联的漏洞弧段的总严重程度成正比。因此,通过分析 $D(x)$ 簇首数量,可以判定该区域的漏洞弧段分布区域,确定节点感知范围的盲区情况。

漏洞弧段可用于检测通信网络安全漏洞,以通信网络节点为依据计算出安全漏洞弧序列^[11]。节点在通信网络中的相对位置关系如图1所示。

以图1中的节点 a 为例,将通过节点 a 的水平和垂直射线分别设为 X 轴与 Y 轴。以 X 轴和 Y 轴为依据,能计算出节点 a 与其他节点间的距离。但仅根据距离不能计算节点的准确位置,因此,所提方法采用虚拟移动的方式确定节点间的相对位置信息^[12]。将节点 a 沿 X 和 Y 轴向另一个节点虚拟移动一定距离后形成新线段,其与原线段的夹角分别为 ρ 和 α ,如图2所示。

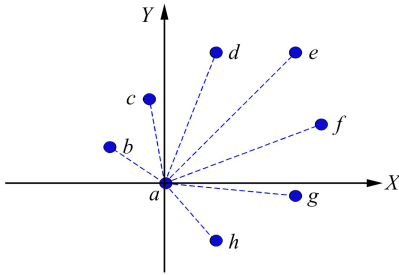


图1 节点相对位置

Fig. 1 Relative position of nodes

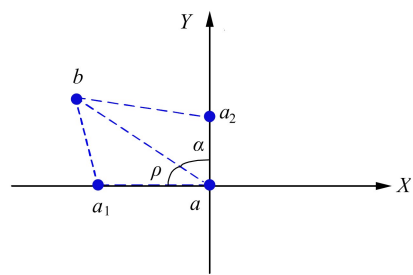


图2 相邻节点的相对位置关系

Fig. 2 Relative position relationship of adjacent nodes

据图2可知,节点 a 与 b 的相对位置根据虚拟移动的2个虚拟节点确定,设节点 a 与 b 的位置信息是 $\langle d_1, \rho \rangle$, d_1 表示2节点的相对距离,夹角 ρ 与 α 的关系式为

$$\rho - \alpha = \pi/2. \quad (10)$$

根据式(10)可以计算出各个节点在通信网络中的位置,设 r 代表节点在网络中的感知半径,通过

$$L_{ij} = \arccos(d_{ij}/2r) \times 2 \quad (11)$$

确定两个相邻节点的安全漏洞弧长^[13]。其中 d_{ij} 表示相邻节点 i 与 j 在通信网络中的相对距离。以 L_{ij} 为依据计算出安全漏洞弧的方位角信息 $\langle \rho_{ij} - L_{ij}/2, \rho_{ij} + L_{ij}/2 \rangle$ 。据图1可知,各相邻节点的监测范围和节点间的安全漏洞弧之间存在相交的情况,因此可以得到 $d_{ij} \leq 2r$ 的结论^[14]。以 d_{ij} 为依据,未被邻近节点所覆盖安全漏洞的弧序列被定义为漏洞弧段,这些漏洞弧段被应用于检测通信网络安全漏洞。

笔者方法采用蚁群算法完成通信网络节点的漏洞弧段信息的检测^[15]。由于被动分簇算法已将节点划分到不同簇中,蚁群算法在检测时可结合簇结构信息。通过

$$H_{uv}^m(t) = [\beta_{uv}(t)^\varepsilon (Q_v)^\omega l_{uv}(t)^\xi] / [\sum_{z \in Y_u} \beta_{uz}(t)^\varepsilon (Q_z)^\omega l_{uz}(t)^\xi] \quad (12)$$

确定需要遍历的通信网络节点。其中 $H_{uv}^m(t)$ 表示在 t 时刻处于节点 u 处的蚂蚁转移到节点 v 的概率; β_{uv}

表示网络节点 u 和节点 v 二者路径间的信息素浓度; ε 表示信息素对应的加权因子; Q_v 与 Q_z 表示网络节点 u 的邻近节点 v 和 z 的标签值; ω 表示节点标签值的加权因子数值; l_{uv} 表示节点 u 与 v 在通信网络中的直线距离; ζ 表示通信网络节点 u 与 v 直线距离的加权因子; $Y_u \in z$, Y_u 表示节点 u 邻近节点中未被蚁群覆盖的节点合集。在通信网络节点安全漏洞弧的检测过程中,蚁群通常会选择遍历标签值较大的节点,同时考虑簇内节点间的相互关系和通信模式,将信息素浓度作为依据,判断安全漏洞的实际位置。漏洞检测过程如下:

$$\beta_{uv}(t+1) = (1 - \sigma)\beta_{uv}(t) + M,$$

(13)

其中 σ 表示在蚂蚁信息素中的挥发系数; M 表示节点 u 与 v 路径信息素的变化量。通过对通信网络节点的蚁群遍历活动分析发现,具有较高信息素浓度的节点会形成最小周期环,这些环即对应通信网络中的所有安全漏洞弧段。由于采用了被动分簇算法,蚁群算法在簇结构内进行检测,能更高效地定位漏洞位置。因此,当网络节点被蚂蚁完全遍历覆盖后可以准确获得漏洞所处的位置,进而结合簇结构信息全面评估网络的安全状况,实现通信网络安全漏洞检测。

基于蚁群算法进行网络安全漏洞检测的伪代码如下:

```
text
ALGORITHM NetworkSecurityAssessment

INPUT:
network_topology,
num_ants = 50,
max_iterations = 200,
alpha = 1,
beta = 2,
gamma = 1,
rho = 0.1, // 信息素挥发系数
Q = 1.5 // 信息素增强系数

OUTPUT: security_assessment

BEGIN
// Phase 1: 网络预处理
clusters ← PASSIVE_CLUSTERING( network_topology )
vulnerability_arcs ← CALCULATE_VULNERABILITY_ARCS( clusters )

// Phase 2: 初始化信息素矩阵
pheromone_matrix ← INITIALIZE_PHEROMONE( network_topology )
best_path ← NULL
best_score ← ∞

// Phase 3: 蚁群优化主循环
FOR iter FROM 1 TO max_iterations DO
  FOR EACH ant IN ants DO
    // 3.1 蚂蚁路径构建
    current_node ← SELECT_START_NODE()
    path ← [ current_node ]
    unvisited_nodes ← ALL_NODES \ { current_node }

    WHILE unvisited_nodes ≠ ∅ DO
```

```
next_node←SELECT_NEXT_NODE(  
current_node,  
unvisited_nodes,  
pheromone_matrix,  
vulnerability_arcs,  
alpha, beta, gamma  
)  
path.APPEND( next_node)  
unvisited_nodes.REMOVE( next_node)  
current_node←next_node  
END WHILE  
  
//3.2 评估路径质量  
path_score←EVALUATE_PATH( path, vulnerability_arcs)  
IF path_score < best_score THEN  
best_score←path_score  
best_path←path  
END IF  
END FOR  
  
//3.3 信息素更新  
UPDATE_PHEROMONE( pheromone_matrix, best_path, Q, rho)  
  
//3.4 收敛检查  
IF HAS_CONVERGED( pheromone_matrix) THEN  
BREAK  
END IF  
END FOR  
  
//Phase 4: 结果分析  
min_cycle_rings←FIND_MIN_CYCLE_RINGS( best_path)  
vulnerability_segments←EXTRACT_VULNERABILITY_SEGMENTS( min_cycle_rings)  
security_assessment←EVALUATE_NETWORK_SECURITY( vulnerability_segments)  
  
RETURN security_assessment  
END ALGORITHM  
  
//辅助函数：选择下一个节点  
FUNCTION SELECT_NEXT_NODE( current, candidates, pheromone, vulnerabilities,  $\alpha, \beta, \gamma$ )  
probabilities←[ ]  
total←0  
  
FOR EACH node IN candidates DO  
// 计算选择概率  
 $\tau$ ←pheromone[ current][ node] //信息素  
 $\eta$ ←CALCULATE_HEURISTIC( node, vulnerabilities) //启发式信息  
 $\delta$ ←CALCULATE_DISTANCE( current, node) //距离因子
```

```
probability←(  $\tau^{\alpha}$  ) * (  $\eta^{\beta}$  ) * (  $\delta^{\gamma}$  )
probabilities. APPEND( probability )
total←total+probability
END FOR

//归一化概率
FOR i FROM 0 TO LENGTH( probabilities )-1 DO
probabilities[ i ]←probabilities[ i ]/total
END FOR

RETURN ROUlette_WHEEL_SELECTION( candidates, probabilities )
END FUNCTION

//辅助函数: 更新信息素
FUNCTION UPDATE_PHEROMONE( pheromone, best_path, Q,  $\rho$  )
//信息素挥发
FOR EACH i, j IN pheromone DO
pheromone[ i ][ j ]←( 1- $\rho$  ) * pheromone[ i ][ j ]
END FOR

//信息素增强( 只在最优路径上 )
FOR k FROM 0 TO LENGTH( best_path )-2 DO
i←best_path[ k ]
j←best_path[ k+1 ]
pheromone[ i ][ j ]←pheromone[ i ][ j ]+Q/CALCULATE_PATH_LENGTH( best_path )
END FOR
END FUNCTION
```

2 实验与分析

为验证通信网络节点攻击漏洞安全检测方法的整体有效性, 需要对其进行实验测试, 其测试的通信网络拓扑结构如图 3 所示。

图 3 中, 硬件方面, 云端采用高性能云服务器集群, 配备多核处理器与大容量存储; 路由器为企业级, 支持多种路由协议; 服务器分应用与数据库服务器, 应用服务器运行测试程序, 数据库服务器存储数据; 主机用高端工作站运行检测工具; 交换机为多层, 可灵活管理流量; 10 台 PC 机模拟用户终端。软件方面, 云端和服务器装 Linux 系统, 主机和 PC 机装 Windows 系统, 应用服务器部署自定义 Web 应用, 数据库用 MySQL。网络配置采用静态 IP, 划分不同子网, 路由器配置静态路由实现互联, 交换机划分 VLAN(Virtual Local Area Network)并设置 ACL(Access Control List)限制非法访问, 以此构建真实可控的测试环境。

实验参数设置中, 构建 6 层结构的 CNN 模型。其中包含 2 个卷积层, 卷积核大小均设置为 3×3, 第 1 个卷积层的卷积核数量为 32, 第 2 个卷积层卷积核数量增加至 64。设置 2 个最大池化层, 池化窗口大小为 2×2。设置 1 个全连接层, 神经元数量为 128。输出层神经元数量为 5 个。在训练过程中, 使用 Adam 优化器, 学习率设定为 0.001, 批次大小为 32, 训练轮数为 500。蚁群算法设置蚂蚁数量为 50 只。

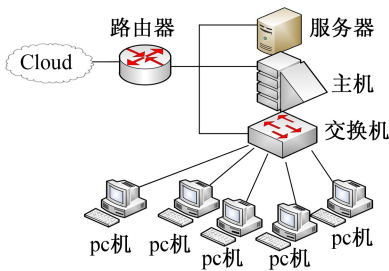


图 3 通信网络拓扑结构

Fig. 3 Communication network topology structure

Fig. 3 Communication network topology structure

蚂蚁在通信网络节点构成的图中进行遍历搜索,迭代次数设定为 200 次。信息素更新策略采用蚁周模型,信息素挥发系数设置为 0.1,信息素增强系数为 1.5。

为确保实验的严谨性与可复现性,笔者采用公开数据集与自建仿真数据集相结合的方式验证。实验构建了包含 200 个节点的复杂网络拓扑,其网络直径量化为 6,平均节点度为 3.5,并动态选举产生 20 个簇首节点以模拟真实通信环境。针对资源耗尽、协议栈漏洞、路由欺骗等 5 类典型网络攻击,采用自相似模型生成背景流量,并基于泊松过程注入攻击流量。通过 OMNeT++ 平台进行 24 h 高保真仿真,精确记录节点资源利用率、链路质量及路由事件等细粒度数据。实验全面评估了所提方法在检测精度、效率及覆盖率等方面的性能,并与主流方法进行对比,验证其优越性。

由于通信网络中的节点数量较多,为缩小检测范围,提高检测效率,对网络中的异常节点进行检测,在确定的异常节点中检测攻击漏洞。通信网络节点攻击漏洞安全检测方法的异常节点检测结果如图 4 所示。

分析图 4 可知,所提方法在异常节点的识别数量上,没有遗漏异常节点,精准定位网络中存在的所有异常节点,表明该方法具有极高的敏感性,不会遗漏潜在的安全隐患。在识别位置方面同样精准无误,为后续的漏洞检测提供精确的目标指引。得到优异结果是因为该方法以事件源分割时间窗口确定相关系数,并且其在固定监测时间窗口内遍历滑动,能全面记录和解析节点的空间相关特征向量。通过对历史时间片段的重新规划,深入挖掘节点在不同时间段的特征变化,从而获取更为准确和全面的节点特征。基于该特征,实现对异常节点的高质量检测。

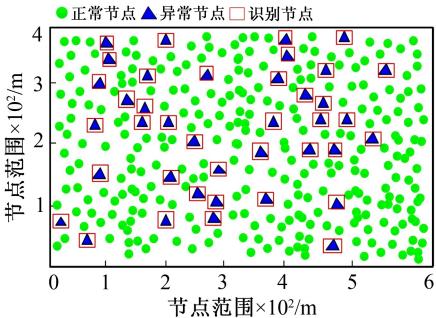


图 4 异常节点检测结果

Fig. 4 Abnormal node detection results

设计消融实验,验证笔者提出的通信网络异常节点检测方法中各个关键组件,即基于滑动窗口互相关系数与时间模糊性的特征提取方法、轻量化 CNN 模型与局部特征降维策略、基于蚁群优化与被动分簇的漏洞安全检测方法对整体检测性能的贡献,明确各组件在提升检测精度、效率以及缩小漏洞检测范围等方面的作用。设置基准组为按照笔者提出的完整方法,首先使用基于滑动窗口互相关系数与时间模糊性的特征提取方法提取节点特征,然后使用轻量化 CNN 模型与局部特征降维策略进行异常节点检测,最后使用基于蚁群优化与被动分簇的漏洞安全检测方法对异常节点进行漏洞检测。设置特征提取方法消融组为采用传统特征提取方法提取节点特征,其余步骤与基准组相同,记录相应指标。设置 CNN 模型消融组为使用传统全局特征学习 CNN 模型进行异常节点检测,其余步骤与基准组相同,记录相应指标。设置漏洞检测方法消融组为使用传统漏洞检测方法对异常节点进行漏洞检测,其余步骤与基准组相同,记录相应指标。对比各实验组与基准组的实验指标,分析各关键组件对整体检测性能的影响。结果如表 1 所示。

表 1 消融实验结果对比

Tab. 1 Comparison of ablation results

实验组别	准确率	召回率	F_1 值	检测时间/ms	模型训练时间/ms	漏洞检测覆盖率
基准组	0.95	0.92	0.93	120.5	3 612	0.88
特征提取方法消融组	0.90	0.85	0.87	230.2	3 628	0.80
CNN 模型消融组	0.88	0.82	0.85	245.7	4 229	0.78
漏洞检测方法消融组	0.93	0.90	0.91	225.3	3 624	0.75

分析表 1 可知,在检测精度方面,基准组准确率、召回率和 F_1 值均最高。特征提取方法消融组采用传统方法后,3 项指标均下降,说明基于滑动窗口互相关系数与时间模糊性的特征提取方法能更有效捕捉节点特征,提升检测准确性。CNN 模型消融组使用传统全局特征学习 CNN 模型,检测精度指标有所降低,表明轻量化 CNN 模型与局部特征降维策略在提高模型对异常节点识别能力上更具优势。漏洞检测方法消融组

指标同样下滑,体现了基于蚁群优化与被动分簇的漏洞安全检测方法在精准定位漏洞上的有效性。在检测效率方面,基准组检测时间和模型训练时间最短。消融组在这两方面时间均增加,说明各关键组件有助于提高检测效率,减少时间成本。在漏洞检测范围方面,基准组漏洞检测覆盖率最高。消融组覆盖率降低,表明各组件协同作用能更全面地检测出网络中存在的漏洞。综上,笔者提出的各关键组件对提升通信网络异常节点检测的整体性能,包括检测精度、效率和漏洞检测范围,均起到重要作用。

为体现所提方法的优越性,设计对比实验。采用所提方法、基于门控图神经网络的检测方法^[4]与基于图神经网络的检测方法^[5]对节点攻击漏洞进行检测,结果如图5所示。

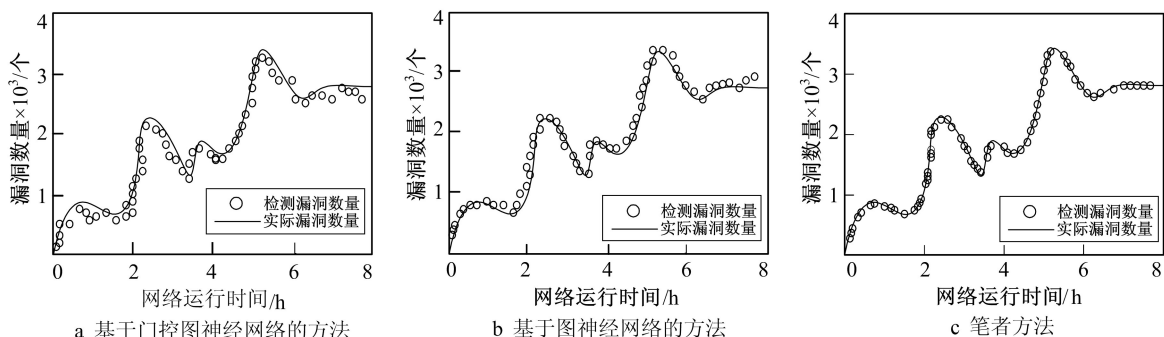


图5 各方法漏洞检测结果

Fig.5 Vulnerability detection results of each method

分析图5可知,所提方法精准检测出了该时段内几乎所有的节点攻击漏洞,检测数量与实际值高度吻合,准确率达99.2%,表明所提方法在该时间段对漏洞的捕捉极为敏锐且全面。而基于门控图神经网络的方法存在一定漏报,部分潜在漏洞未被检出,说明其对某些特定类型或特征的漏洞挖掘能力不足;基于图神经网络的方法漏报情况更为明显,挖掘能力相对更弱,难以有效识别该时段内的部分关键漏洞。综上,所提方法保持了稳定且出色的检测性能,能准确找出各时间段的漏洞,凸显了所提方法的优越性。这是因为所提方法借助蚁群算法检测漏洞弧段信息,以此确定遍历节点,依据信息素浓度判断漏洞位置,为通信网络安全提供了坚实保障。

F_1 分数是精确率和召回率的调和平均值,其数值能确定漏洞检测的实际水平,反映模型的整体情况。 F_1 数值与漏洞检测水平成正比,如下:

$$F_1 = 2PR/(P + R), \quad (14)$$

其中 P 表示漏洞样本中被判断为正确样本的准确率; R 表示召回率,即被成功检测出的漏洞样本占有所有漏洞样本的比例。图6给出了各方法 F_1 数值。

分析图6可知,所提方法在 F_1 数值指标上具有显著优势,其数值平均为0.8。相较于其他两种方法,所提方法结合被动分簇算法,考虑簇内节点关系和通信模式,能更全面地分析网络漏洞。该方法在面对复杂通信网络时,可综合考虑多种因素,提高漏洞检测的准确性与全面性。而其他方法仅依靠 CNN 局部特征学习与 Softmax 分类识别异常节点以及滑动时间窗口分割与时空互相关分析提取低维可解释节点特征,虽能在一定程度上提升检测效果,但无法从整体上精准定位和全面检测漏洞,因此在 F_1 数值指标上优势不明显。

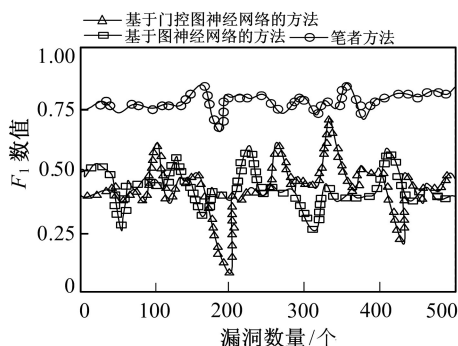


图6 各方法 F_1 数值

Fig.6 F_1 values for each method

笔者设计了3种不同规模的网络拓扑,小规模100节点、中规模300节点、大规模600节点。选取3种具有代表性的漏洞进行测试,包括资源耗尽、路由欺骗。通过 F_1 -Score、检测覆盖率验证泛化性,通过平均检测延迟、算法收敛时间验证时效性。结果如表2所示。

表2 不同方法在不同网络规模与漏洞类型下的性能与效率对比

Tab.2 Performance and efficiency comparison of different methods under different network sizes and vulnerability types

网络规模	漏洞类型	评估指标	所提方法	基于门控图神经网络的方法	基于图神经网络的方法
小规模(100 节点)	资源耗尽	F_1 -Score/%	98.5	95.2	92.1
		检测覆盖率/%	99.1	96.5	94
		平均检测延迟/ms	125	180	210
		算法收敛时间/s	8.5	10.2	12.8
	路由欺骗	F_1 -Score/%	97.8	90.1	85.4
		检测覆盖率/%	98.5	92	88.5
		平均检测延迟/ms	155	350	450
		算法收敛时间/s	9.1	11.5	14.2
	数据篡改	F_1 -Score/%	96.8	89.5	83.2
		检测覆盖率/%	97.5	90	85.8
		平均检测延迟/ms	230	520	680
		算法收敛时间/s	35.2	65.8	88.5
中规模(300 节点)	资源耗尽	F_1 -Score/%	95.5	82.3	75.6
		检测覆盖率/%	96	85.1	80.2
		平均检测延迟/ms	280	650	850
		算法收敛时间/s	38.7	70.1	95.3
	路由欺骗	F_1 -Score/%	94.2	78.8	70.5
		检测覆盖率/%	95.8	82.5	78
		平均检测延迟/ms	450	1 250	1 650
		算法收敛时间/s	125.5	285.4	超时(>600)
	数据篡改	F_1 -Score/%	92.1	75.6	65.8
		检测覆盖率/%	93.5	78.9	72.1
		平均检测延迟/ms	520	1 550	1 950
		算法收敛时间/s	132.8	310.2	超时(>600)
大规模(600 节点)	资源耗尽	F_1 -Score/%			
		检测覆盖率/%			
		平均检测延迟/ms			
		算法收敛时间/s			
	路由欺骗	F_1 -Score/%			
		检测覆盖率/%			
		平均检测延迟/ms			
		算法收敛时间/s			
	数据篡改	F_1 -Score/%			
		检测覆盖率/%			
		平均检测延迟/ms			
		算法收敛时间/s			

分析表2可知,所提方法在不同网络规模与漏洞类型下均表现出卓越的综合性能。在泛化性方面,所提方法的 F_1 -Score与检测覆盖率在所有测试场景下均显著优于两种对比方法,尤其在大规模网络中对路由欺骗漏洞的检测中, F_1 -Score领先基于图神经网络的方法超过26%,表现出强大的场景适应能力。在时效性方面,所提方法的优势更为突出,其平均检测延迟仅为对比方法的1/3至1/4,算法收敛时间更是大幅缩短,在大规模网络中能稳定完成检测而对比方法已出现超时。上述充分证明,通过被动分簇实现的局部精细化检测与蚁群算法对漏洞弧段的定向搜索机制,有效克服了传统图神经网络方法在全局计算上的效率瓶颈,实现了检测精度与效率的平衡,为大规模复杂网络的实时安全防护提供了可靠解决方案。

3 结 语

笔者提出通信网络节点攻击漏洞安全检测方法,借助动态时空特征融合与蚁群优化算法,依据安全漏洞弧长和信息素浓度精准定位漏洞位置,实现通信网络安全漏洞检测。实验表明,该方法 F_1 值较其他方法提升1.5%~8%,能精准识别漏洞、缩小检测范围,为网络运维提供高效安全保障,具有实际应用价值。未来,将探索多源异构数据融合的动态时空特征建模,结合自适应图神经网络架构提升网络适应性,并研究漏洞检测与智能修复一体化机制,推动通信网络防御智能化、主动化发展。

参考文献:

[1] 刘昭, 邹权臣, 于恬, 等. 一种约束制导的机器学习框架漏洞检测方法 [J]. 计算机学报, 2024, 47(5): 1120-1137.
LIU Z, ZOU Q C, YU T, et al. Constraint-Guided Vulnerability Detection Techniques for Machine Learning Framework [J].

- Chinese Journal of Computers, 2024, 47 (5): 1120-1137.
- [2] 何清林, 王丽宏, 陈艳姣, 等. 一种自动实时的物联网在野漏洞攻击检测方法 [J]. 北京航空航天大学学报, 2024, 50 (7): 2195-2205.
- HE Q L, WANG L H, CHEN Y J, et al. An Automatic and Real-Time Detection Method of IoT in-the-Wild Vulnerability Attack [J]. Journal of Beijing University of Aeronautics and Astronautics, 2024, 50(7): 2195-2205.
- [3] 杨宏宇, 杨海云, 张良, 等. 基于特征依赖图的源代码漏洞检测方法 [J]. 通信学报, 2023, 44(1): 103-117.
- YANG H Y, YANG H Y, ZHANG L, et al. Feature Dependence Graph Based Source Code Loophole Detection Method [J]. Journal on Communications, 2023, 44(1): 103-117.
- [4] YANG J, RUAN O, ZHANG J X. Tensor-Based Gated Graph Neural Network for Automatic Vulnerability Detection in Source Code [J/OL]. Software Testing, Verification And Reliability, 2024, 34(2): e1867 (2023-11-27) [2025-06-05]. <https://doi.org/10.1002/stvr.1867>.
- [5] 胡雨涛, 王溯远, 吴月明, 等. 基于图神经网络的切片级漏洞检测及解释方法 [J]. 软件学报, 2023, 34(6): 2543-2561.
- HU Y T, WANG S Y, WU Y M, et al. Slice-level Vulnerability Detection and Interpretation Method Based on Graph Neural Network [J]. Journal of Software, 2023, 34(6): 2543-2561.
- [6] CHEN J, YIN Y, CAI S, et al. iGnnVD: A Novel Software Vulnerability Detection Model Based on Integrated Graph Neural Networks [J/OL]. Science of Computer Programming, 2024, 238 (2024-01-15) [2025-06-05]. <https://doi.org/10.1016/j.scico.2023.103156>.
- [7] 刘彦北, 马夕然, 王雯. 可信的图神经网络节点分类方法 [J]. 天津工业大学学报, 2024, 43(1): 82-88.
- LIU Y B, MA X R, WANG W. Node Classification Method Based on Trusted Graph Neural Network [J]. Journal of Tiangong University, 2024, 43(1): 82-88.
- [8] 李云峰, 高云鹏, 张蓬鹤, 等. 多目标优化时域卷积神经网络的窃电行为高准确检测算法 [J]. 电网技术, 2024, 48 (8): 3449-3458.
- LI Y F, GAO Y P, ZHANG P H, et al. High-Accuracy Detection Algorithm of Electricity Theft Behavior Based on Multi-Objective Optimization Time-Domain Convolutional Neural Network [J]. Power System Technology, 2024, 48(8): 3449-3458.
- [9] 张宇娇, 赵志涛, 徐斌, 等. 基于 U-net 卷积神经网络的电磁场快速计算方法 [J]. 电工技术学报, 2024, 39(9): 2730-2742.
- ZHANG Y J, ZHAO Z T, XU B, et al. Fast Calculation Method of Electromagnetic Field Based on U-Net Convolutional Neural Network [J]. Transactions of China Electrotechnical Society, 2024, 39(9): 2730-2742.
- [10] 王园淋, 欧阳勇, 高榕, 等. 基于虚拟节点策略的自监督对比学业预警模型 [J]. 计算机工程与设计, 2024, 45(7): 2195-2202.
- WANG Y L, OUYANG Y, GAO R, et al. Self-Supervised Comparative Academic Early Warning Model Based on Virtual Node Strategy [J]. Computer Engineering and Design, 2024, 45(7): 2195-2202.
- [11] 孔维立, 王峰, 周平华, 等. 改进蚁群算法的无人机三维路径规划 [J]. 电光与控制, 2023, 30(3): 63-69.
- KONG W L, WANG F, ZHOU P H, et al. Three-Dimensional Path Planning of UAVs Based on Improved Ant Colony Algorithm [J]. Electronics Optics & Control, 2023, 30(3): 63-69.
- [12] 李涛, 赵宏生. 基于进化蚁群算法的移动机器人路径优化 [J]. 控制与决策, 2023, 38(3): 612-620.
- LI T, ZHAO H S. Path Optimization for Mobile Robot Based on Evolutionary Ant Colony Algorithm [J]. Control and Decision, 2023, 38(3): 612-620.
- [13] 汪媛媛, 冯学晓. 基于蚁群算法的多跳无线通信数据传输优化 [J]. 网络安全和信息化, 2025(3): 112-114.
- WANG Y Y, FENG X X. Optimization of Multi Hop Wireless Communication Data Transmission Based on Ant Colony Algorithm [J]. Network Security and Informatization, 2025(3): 112-114.
- [14] 谭明, 赵薪. 基于动态自适应蚁群算法的大型无人机智能巡检路径规划系统设计 [J]. 计算机测量与控制, 2025, 33 (5): 337-350.
- TAN M, ZHAO X. Design of Intelligent Inspection Path Planning System for Large UAVs Based on Dynamic Adaptive Ant Swarm Algorithm [J]. Computer Measurement & Control, 2025, 33(5): 337-350.
- [15] 蔡军, 钟志远. 改进蚁群算法的送餐机器人路径规划 [J]. 智能系统学报, 2024, 19(2): 370-380.
- CAI J, ZHONG Z Y. Path Planning of a Meal Delivery Robot Based on an Improved Ant Colony Algorithm [J]. CAAI Transactions on Intelligent Systems, 2024, 19(2): 370-380.