基于Stackelberg博弈和DQN的多类型蜜罐部署方案

韩雨[1]，陈元恒[2]，王一川[1]，马艺宾[1]，黑新宏[1]
（1.西安理工大学计算机科学与工程学院，陕西 西安 710048；2.郑州大学网络空间安全学院，河南 郑州 450001）

摘　要：针对传统蜜罐部署方案在面对日益复杂的网络环境时存在动态适应性差、诱捕能力不足等问题，基于CIC-IDS-2017攻击数据集，提出了一种基于Stackelberg博弈和深度Q网络（DQN）的多类型动态蜜罐部署方案。首先，通过对攻击行为时间–状态建模捕捉攻击行为的时序演化特征，结合马尔可夫预测实现对未知攻击的预判。其次，根据不同蜜罐（低交互、中交互、高交互和拟态蜜罐）的部署成本和诱捕能力的差异性，设计融合攻防效益的综合效用函数。最后，通过Stackelberg博弈主导角色动态切换与DQN策略优化，实现固定资源约束下的最优部署，进一步提升策略的动态适应性。仿真结果表明，所提方案能够有效应对攻击行为的时序演变状态，并在固定资源约束下给出最优的蜜罐部署方案，提升了防御系统的自适应性。此外，该方案对时序攻击的诱捕成功率达96%（在拟态蜜罐情况下），防御效用较传统方案提升35%，且能动态适应多类型攻击场景。

关键词：网络安全；蜜罐诱捕；动态部署；动态博弈；深度Q网络
中图分类号：TN92
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2026025

Multi-type honeypot deployment scheme based on Stackelberg game and DQN

Han Yu[1], Chen Yuanheng[2], Wang Yichuan[1], Ma Yibin[1], Hei Xinhong[1]
1. School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China
2. School of Cyberspace Security, Zhengzhou University, Zhengzhou 450001, China

Abstract: Traditional honeypot deployment schemes suffer from issues such as poor dynamic adaptability and insufficient trapping capability when confronting increasingly complex network environments. Based on the CIC-IDS-2017 attack dataset, a multi-type dynamic honeypot deployment scheme was proposed integrating the Stackelberg game and deep Q-network (DQN). First, by conducting time-state modeling on attack behaviors to capture their temporal evolutionary characteristics, and combining this modeling with Markov prediction, the prediction of unknown attacks was achieved. Secondly, considering the differences in deployment costs and trapping capabilities among different types of honeypots (low-interaction, medium-interaction, high-interaction, and mimic honeypots), a comprehensive utility function that integrated offensive and defensive benefits was designed. Finally, through the dynamic switching of the leading role in the Stackelberg game and DQN-based strategy optimization, optimal deployment under fixed resource constraints was realized, which further enhanced the dynamic adaptability of the strategy. Simulation results demonstrate that the proposed scheme can effectively cope with the temporal evolution of attack behaviors, provide an optimal honeypot deployment scheme under fixed resource constraints, and improve the adaptability of the defense system. Specifically, the scheme achieves a trapping success rate of 96% for temporal attacks (in the case of mimic honeypots), the defense utility is 35% higher than that of traditional schemes, and it can dynamically adapt to multi-type attack scenarios.

Keywords: network security, honeypot trapping, dynamic deployment, dynamic gaming, DQN

收稿日期：2025–09–08；**修回日期：**2026–01–26
通信作者：王一川，chuan@xatu.edu.cn
基金项目：国家重点研发计划基金资助项目（No.2024YFF1401303）
Foundation Item：The National Key Research and Development Program of China (No.2024YFF1401303)

0 引言

随着互联网规模的不断扩展，网络攻击类型日益多样，如拒绝服务（denial of service, DoS）、分布式拒绝服务（distributed denial of service, DDoS）、Web攻击等，攻击手法复杂多变，传统被动式防御体系（如入侵检测[1]、防火墙[2]、恶意代码扫描[3]、网络监控技术[4]等）难以应对复杂多样的网络环境。蜜罐作为一种主动安全防御技术[5]，通过模拟真实网络系统，利用云技术或者虚拟化技术诱导攻击行为，保护真实资产，能够有效延缓攻击进程，为防御者收集情报和分析攻击者行为提供了有效支撑，是网络安全主动防御体系的重要组成部分。

近年来，学术界关于蜜罐技术的研究与应用已经取得了显著成效，但仍存在一些不足。早期研究主要集中于创建多层欺骗环境以提高诱捕能力。例如，文献[6]提出了蜜罐防御有效性的安全测量方法，用于评估不同部署策略对“蜜罐”防御策略的有效性和风险。然而，这种方法很难应对快速变化的攻击方法和网络环境。文献[7]设计了高交互蜜罐系统，密切观察攻击者的行为，在主动防御、信息收集、模块化设计和易用性等方面具有显著优势，但实施难度大，误报率和漏报率高。文献[8]介绍了蜜罐技术在工业控制系统中的应用，以增强关键基础设施的安全性，但其在高流量场景下扩展性不足。文献[9]提出的拟态防御理论虽是创新性防御思路，但未与蜜罐技术深度融合。

针对传统静态蜜罐的局限性，文献[10]提出了物联网蜜罐动态博弈模型，该模型侧重于网络连接变化引起的状态演化，但仅针对物联网（Internet of things, IoT）场景，通用性不足。文献[11]基于演化博弈理论优化蜜罐部署防御策略的方法，提高了系统安全性。然而，该方法过于依赖模型假设，缺乏真实数据支撑。文献[12]提出了一种基于多目标遗传算法的蜜罐部署优化方法，该方法考虑了部署节点的重要性和系统的资源开销，但网络模型结构较为静态，难以适应动态攻击环境。文献[13]提出了一种动态蜜罐切换机制，拦截持续性攻击具有一定的时效性，但缺乏策略演化能力，无法应对持续变异的攻击。文献[14]提出了一种基于对抗博弈论建模的物联网安全外壳（secure shell, SSH）自适应蜜罐策略，以提高蜜罐系统的适应性，但物联网环境设备资源有限导致计算复杂度高、部署和维护困难。文献[15]提出的动态软件定义网络（software defined networking, SDN）蜜罐可以动态响应不同的攻击阶段，并通过调整蜜罐配置和行为来有效应对攻击者的多阶段攻击行为，但蜜罐类型单一固定，无法适应多类型攻击。

同时，DQN算法与博弈论相结合的方法在蜜罐研究中也得到了广泛关注。文献[16]利用DQN算法优化蜜罐部署策略，提升了策略的动态适应性，但未考虑攻防双方的角色变换。文献[17]通过Stackelberg博弈决策方法进行蜜罐部署，但未结合真实攻击数据的时序演化特征。文献[18]提出了基于马尔可夫决策过程（Markov decision process, MDP）的蜜罐调度方法，可在有限资源下优化部署，但未考虑多类型蜜罐的差异性。

综上所述，当前针对蜜罐的研究虽然在优化建模和策略调度方面有了显著进展，但缺乏对攻击行为演化的动态建模，策略演化能力不足。因此，针对未知攻击的自适应蜜罐生成，多类型蜜罐动态部署的研究具有非常重要的理论意义。

因此，本文基于CIC-IDS-2017攻击数据集，结合Stackelberg博弈和DQN算法实现多类型蜜罐动态部署与优化。本文主要贡献如下。

1) 通用性和动态适应性的提升：融合CIC-IDS-2017攻击数据集与马尔可夫预测，构建时序化Stackelberg博弈模型，引入未知攻击触发攻防角色切换，通过DQN算法优化策略，降低对模型假设的依赖，实现效用驱动的自适应演化。

2) 多类型蜜罐最优调度：定义“最优蜜罐部署”，针对低交互、中交互、高交互和拟态蜜罐的部署成本和诱捕能力的不同，设计基于攻防效益、诱捕收益和攻防成本的综合效用函数，实现最优蜜罐部署。

3) 性能优势显著：实验证明，本文方案具有很好的动态适应性，在动态攻击环境下的策略更新时延小于或等于5 s，资源利用率达到92%，且能通过拟态蜜罐自动化部署实现对真实资产的隔离防护。

1 系统模型

本文首先对CICIDS2017攻击数据集进行预处理，该数据集涵盖了多种真实网络环境下的攻击行

为，包括DoS、DDoS、Web攻击、端口扫描和暴力破解等多种攻击类型，具有高度的真实性与代表性，通过归纳数据集中的相关攻击类型和频率，并结合攻击的时间序列，对攻击行为进行时间–状态建模，伴随着防御者多类型蜜罐集群的部署，需要借助Stackelberg博弈模型和强化学习算法，不断优化自身效用函数。同时，根据量化不同类型组合蜜罐的部署成本和诱捕能力，优化部署策略，进而获得最优蜜罐部署方案。关于相关研究，本节首先介绍以下几个核心思想。

1.1　数据预处理

为了消除原始流量数据的量纲差异并适应DQN模型的输入要求，本文剔除了CIC-IDS-2017攻击数据集中包含缺失值和无限值的无效样本，防止梯度爆炸或计算错误，去除源/目的IP地址、时间戳等无法表征攻击行为模式的标识性特征，保留流持续时间、包长度统计量等78维流特征。目的在于原始数据集中所要剔除的特征易导致模型对特定网络拓扑的过拟合。因此，本文剔除上述6类标识性特征，保留描述流量统计行为的78维流特征，包括流持续时间、包长度统计量、标志位计数、包到达间隔（inter-arrival time, IAT）等，确保模型专注于学习攻击行为的本质模式而非特定环境参数。

同时，本文采用最大最小归一化方法将所有连续型特征映射至[0,1]，以加速神经网络收敛，计算式为

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{1}$$

其中，x为原始流量样本中的具体特征值，$x_{\min}$为该维特征在整个训练数据集中的统计最小值，$x_{\max}$为该维特征在整个训练数据集中的统计最大值，x'为经过线性映射处理后的归一化特征值，其取值范围严格约束在[0,1]，消除了不同特征量纲差异对梯度下降方向的影响。

在标签编码方面，本文对攻击类型标签进行One-Hot独热编码，将多分类标签转化为向量形式。考虑到网络流量的时序依赖性，本文未采用随机打乱划分，而是按照时间序列顺序，将预处理后的数据集按8:1:1的比例划分为训练集、验证集和测试集。其中训练集用于DQN智能体的策略学习，验证集用于超参数调优，测试集用于评估模型在未知流量环境下的泛化能力。核心参数设置如表1所示。

表1　　核心参数设置

参数名称	参数值	操作系统参数说明
优化器	Adam	采用自适应矩估计进行梯度下降
攻击机初始学习率	0.001	权重更新步长，控制收敛速度
批次大小	64	每次参数更新所使用的样本数量
经验回放池容量	10 000	存储历史(s, a, r, s')四元组的队列长度
折扣因子	0.90	用于平衡当前奖励与未来长期回报的权重
探索率	1.0 → 0.01	初始为全探索，随训练步数线性衰减至0.01
目标网络更新频率	200 步	每训练200步同步一次主网络参数
损失函数	MSE Loss	均方误差损失，用于衡量Q值预测偏差

CIC-IDS2017攻击数据集预处理结果如表2所示。

表2　CIC-IDS2017攻击数据集预处理结果

标签类型（良性、攻击）	标签数量/个	特征维度
BEGIN	2 271 320	78
FTP-Patator	7 935	78
SSH_Patator	5 897	78
DoS-Hulk	2 301 24	78
DoD-GoldenEye	1 029 3	78
DoS-slowloris	5 796	78
DoS-Slowhttptest	5 499	78
Heartbleed	11	78
Web_Attack-Brute_Force	1 507	78
Web_Attack-XSS	652	78
Web_Attack-SQL_Injection	21	78
Infiltration	36	78
Bot	1 956	78
PortScan	1 588 04	78
DDoS	1 280 25	78

1.2　符号体系

为确保模型描述的严谨性，本节对系统中的核心符号、策略空间和状态空间进行统一的数学定义，如表3所示。

表3 核心符号定义

符号	维度/类型	含义	备注
m,n	标量	攻击类型数量，蜜罐类型数量	本文设m=8，n=4
k	标量	时间滑动窗口大小	本文设k=5
Δ_m,Δ_n	集合	攻击策略与防御策略的单纯形空间	满足概率和为1且非负
A_t	R^m	t时刻攻击策略向量	$A_t \in \Delta_m$
D_t	R_t^n	t时刻蜜罐部署策略向量	$D_t \in \Delta_n$
S_t	$R^{k(m+n)}$	t时刻系统状态（级联向量）	包含历史k步攻防序列
h_t	R^{64}	状态嵌入向量	DQN输入层特征
M	$R^{n\times m}$	诱捕效能矩阵	M_{ij}为蜜罐i针对攻击j的效能
R_d,R_a	标量	防御与攻击的总资源预算	约束条件常数
C_d,C_a	R^n,R^m	单位部署成本向量，单位攻击成本向量	—
γ	标量	折扣因子	用于长期奖励衰减
θ,ϕ	标量	防御与攻击策略网络参数	—
$Q(s,a;\theta)$	函数	Q函数	—
π_θ,π_ϕ	函数	防御与攻击策略函数	—

1.3 时间–状态空间建模

1.3.1 攻防策略定义

攻击者策略。攻击者采用8类攻击（SQL注入攻击、命令注入攻击等，m=8），策略向量$\boldsymbol{A}_t=\left[a_{t,1},a_{t,2},\cdots,a_{t,n}\right]^{\mathrm{T}}\in\Delta^m$，其中$a_{t,i}$表示第$i$类攻击的资源分配比例，满足$\sum_{i=1}^{m}a_{t,i}=1$且$a_{t,i}>0$。

防御者策略。防御者部署低交互、中交互、高交互和拟态蜜罐4类蜜罐（n=4），策略向量$\boldsymbol{D}_t=\left[d_{t,1},d_{t,2},\cdots,d_{t,n}\right]^{\mathrm{T}}\in\Delta^n$，其中$d_{t,j}$表示第$j$类蜜罐的资源分配比例，满足$\sum_{j=1}^{n}d_{t,j}=1$且$d_{t,j}>0$。

1.3.2 时间–状态空间数学定义

为有效捕捉攻击行为的动态演化过程，将系统的状态表示为一系列历史行为的窗口，每个时间点的状态包含过去k轮攻击行为及响应的蜜罐部署策略。本文将$\boldsymbol{S}_t$定义为一个高维级联向量，如式(2)所示。

$$\boldsymbol{S}_t=\left\{\boldsymbol{A}_{t-k},\boldsymbol{A}_{t-k+1},\cdots\boldsymbol{A}_{t-1},\boldsymbol{D}_{t-k},\boldsymbol{A}_{t-k+1},\boldsymbol{D}_{t-1}\right\} \tag{2}$$

其中，$\boldsymbol{A}_t\in R^m$为时间t时的攻击者行为分布（基于表6中8类攻击的概率向量），$\boldsymbol{D}_t\in R^n$为时间t时的蜜罐部署策略（4类蜜罐的资源分配比例），K=5为实验验证的窗口大小。

状态处理。$\boldsymbol{S}_t$是高维级联向量（5×8+4×4=56维），直接作为DQN算法输入可能导致收敛困难。通过神经网络的嵌入层将其映射为低维特征向量$\boldsymbol{h}_t$，可表示为

$$\boldsymbol{h}_t=f_{\mathrm{embed}}(\boldsymbol{S}_t)\in R^{d_{\mathrm{model}}} \tag{3}$$

其中，d_{model}=64为实验设定的嵌入维度。$\boldsymbol{h}_t$保留了关键的时序关联特征，作为策略函数π_θ的实际输入，需通过DQN的嵌入层将其映射为低维状态嵌入向量$\boldsymbol{h}_t\in R^{64}$，既能保留时序特征，也能降低计算复杂度。

1.3.3 输出蜜罐部署策略

DQN的决策结果是给4类蜜罐分配资源，动作$\boldsymbol{a}_t$是4维向量$[d_{\text{低}},d_{\text{中}},d_{\text{高}},d_{\text{拟态}}]$，其中$d_i\in(0,1)$且$\sum_{i=1}^{4}d_i=1$，每个维度代表对应蜜罐的资源分配比例。

设置奖励机制，引导DQN学习对防御有利的策略，关联Stackelberg博弈中的防御者效用函数，确保优化方向是在最小化部署成本的约束下，最大化诱捕收益。

奖励函数设计：第t步的奖励为$r_t=U_d(\boldsymbol{D}_t,\boldsymbol{A}_t)-\alpha$策略波动惩罚。

α策略波动惩罚：抑制蜜罐部署比例发生剧烈变化，避免资源浪费，公式为$\alpha\|D_t - D_{t-1}\|^2$（$\alpha$=0.1为正则化系数）。

若某部署策略D_t能“高诱捕、低成本和策略平稳”，则r_t为正且数值大，DQN会倾向于保留该策略；反之若r_t为负（如高成本但诱捕差），则DQN会调整策略。

1.4　马尔可夫预测模型

基于CIC-IDS-2017历史攻击数据，采用马尔可夫过程预测攻击行为的转移矩阵，假设攻击者使用当前状态选择的攻击类型，预测下一步的攻击行为，通过训练马尔可夫模型，得到攻击行为的转移矩阵$\boldsymbol{P} \in R^{m\times m}$，用来对未知攻击进行预测，为蜜罐部署策略提供决策依据，转移矩阵的计算式为

$$P_{i,j} = \frac{\boldsymbol{T}_{i,j}}{\sum_{j=1}^{m} T_{i,j}} \tag{4}$$

其中，$\boldsymbol{T}_{i,j}$为历史攻击行为转移的计数矩阵（如SQL注入攻击→命令注入攻击的转移次数），$P_{i,j}$表示从攻击类型i转移到j的概率。该模型可提前0.5~1个时间步预判未知攻击，为蜜罐部署策略提供决策依据。

1.5　Stackelberg博弈模型

1.5.1　角色定义

在斯塔克尔伯格博弈模型中，首先，防御者会部署初始蜜罐集群，包含低交互、中交互、高交互和拟态蜜罐，因此防御者是领导者，而攻击者会通过侦察等操作，根据已部署的蜜罐集群选择相应的攻击策略以绕过蜜罐诱捕，因此攻击者是追随者。但攻击行为具有时序性的动态演化特征，所以攻防博弈模型是动态变化的。在每一轮博弈中，防御方和攻击方的策略都会不断调整，并最终收敛到一个均衡点。

1.5.2　效用函数

攻击者效用函数为

$$U_a(\boldsymbol{D}_t,\boldsymbol{A}_t) = \boldsymbol{A}_t^{\mathrm{T}}(\boldsymbol{R} - \boldsymbol{L}^{\mathrm{T}}\boldsymbol{D}_t) - \boldsymbol{C}_a^{\mathrm{T}}\boldsymbol{A}_t - \beta\|\boldsymbol{A}_t - \boldsymbol{A}_{t-1}\|_2^2 \tag{5}$$

其中，$\boldsymbol{R} \in R^m$为每类攻击的潜在收益，$\boldsymbol{L}^{\mathrm{T}} \in R^{n\times m}$为蜜罐对攻击者的惩罚矩阵，$\boldsymbol{C}_a^{\mathrm{T}} \in R^m$为攻击者采用每种攻击的单位资源消耗，$\beta > 0$为波动敏感系数，引入该项后，攻击者的目标函数变为关于$\boldsymbol{A}_t$的严格凹函数。

防御者效用函数为

$$U_d(\boldsymbol{D}_t,\boldsymbol{A}_t) = \boldsymbol{D}_t^{\mathrm{T}}\boldsymbol{M}\boldsymbol{A}_t - \boldsymbol{C}_d^{\mathrm{T}}\boldsymbol{D}_t - \alpha\|\boldsymbol{D}_t - \boldsymbol{D}_{t-1}\|_2^2 \tag{6}$$

其中，$\boldsymbol{D}_t^{\mathrm{T}}\boldsymbol{M}\boldsymbol{A}_t$为各类蜜罐对不同攻击类型的诱捕收益的加权总和，$\boldsymbol{C}_d^{\mathrm{T}}$为蜜罐部署成本向量，$\boldsymbol{M} \in R^{n\times m}$为蜜罐诱捕效能矩阵，其第$i$行第$j$列元素$M_{ij}$表示第$i$类蜜罐对第$j$类攻击的诱捕效能，$\alpha > 0$为防御策略平滑系数。

1.5.3　诱捕效能矩阵M的定义与计算

式(6)中$\boldsymbol{M}$是链接蜜罐特性与攻击类型的关键矩阵，其元素$M_{i,j}$表示第i类蜜罐对第j类攻击的诱捕效能，计算式为

$$M_{i,j} = \text{蜜罐诱捕能力} \times \text{攻击易感性} \tag{7}$$

诱捕效能矩阵如式(8)所示。

$$\boldsymbol{M} = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1j} \\ m_{21} & m_{22} & \cdots & m_{2j} \\ \vdots & \vdots & & \vdots \\ m_{i1} & m_{i2} & \cdots & m_{ij} \end{bmatrix} \tag{8}$$

$\boldsymbol{M}$的物理意义是量化“蜜罐–攻击”匹配度，为效用函数提供数值支撑，如拟态蜜罐对SQL注入的$M_{4,1}$=0.96×0.72=0.69，意味着该组合可以产生较高的诱捕效益。

1.5.4　Stackelberg均衡存在性与唯一性分析

在Stackelberg博弈中，防御方制定策略$\boldsymbol{D}_t \in \Delta^n$，攻击方观察到$\boldsymbol{D}_t$后选择最优响应$\boldsymbol{A}_t^*(\boldsymbol{D}_t) \in \Delta^m$。博弈的均衡解（$\boldsymbol{D}^*,\boldsymbol{A}^*$）需满足式(9)与式(10)。

$$\boldsymbol{A}^* = \arg\max_{\boldsymbol{A}\in\Delta^m} U_a\left(\boldsymbol{D}^*,\boldsymbol{A}\right) \tag{9}$$

$$\boldsymbol{D}^* = \arg\max_{\boldsymbol{D}\in\Delta^n} U_d\left(\boldsymbol{D},\boldsymbol{A}^*(\boldsymbol{D})\right) \tag{10}$$

定理1　均衡存在性与唯一性。基于式(5)和式(6)定义的效用函数，在资源约束及策略空间满足凸性的前提下，在任意时间步t，该博弈模型存在唯一的Stackelberg均衡解。

证明　条件1。$\Delta^n = \left\{\boldsymbol{D} \in R^n \middle| \sum_{j=1}^{n} \boldsymbol{D}_j = 1, D_j \geqslant 0\right\}$为概率单纯形，是紧致凸集。同理$\Delta^m$亦然。攻防双方的策略空间$\Delta^m$，$\Delta^n$定义为标准单纯形，即满足$\sum p_i = 1, p_i \geqslant 0$。根据泛函分析理论，欧几里得空间中的单纯形属于非空紧致凸集。

条件2。由式(5)和式(6)可知，引入二次惩罚项$\beta\|A_t - A_{t-1}\|_2^2$后，该函数关于决策变量$\boldsymbol{A}_t$的海森矩

阵为负定矩阵，故 U_a 是关于 $\boldsymbol{A}_t$ 的严格凹函数。根据凸优化理论，定义在紧致凸集上的严格凹函数必存在唯一的全局最大值。因此，攻击者的最优响应映射 F: $\boldsymbol{D} \to \boldsymbol{A}^*(\boldsymbol{D})$ 是单值且连续的函数。

条件3。由于 $\boldsymbol{A}^*(\boldsymbol{D})$ 连续，且防御者效用函数 U_d 关于 $\boldsymbol{D}$ 连续，根据 Stackelberg 博弈的存在性定理，在紧致策略空间内必存在至少一个均衡点。结合上述严格凹函数导致的单值响应特性，该均衡点在每个决策周期内是唯一的。证毕。

1.6 多类型蜜罐动态部署目标

本文针对低交互型、中交互型、高交互型和拟态蜜罐等多种蜜罐类型，系统设计了综合效用函数，将攻击者的诱捕能力、部署成本、策略收益等多个因素结合起来，实现多类型蜜罐的动态部署。

虽然定理 1 证明了静态视角下均衡解的存在性，但在实际网络对抗中，状态空间 $\boldsymbol{S}_t$ 随时间动态演化，且攻击者的收益矩阵 $\boldsymbol{R}$ 和转移概率往往是未知的。求解上述均衡点在计算上具有 NP-hard 复杂度，难以满足实时防御需求。

因此，本文将寻找 Stackelberg 均衡的问题转化为 MDP 下的最优策略求解问题，通过 DQN 算法逼近上述理论均衡点。综合考虑诱捕效益与部署成本，防御者的目标是在满足资源约束的前提下，寻找最优策略序列，其中，$U_{\text{trap}}(D_t^i)$ 为第 i 种蜜罐的诱捕效益，$C_{\text{deploy}}(D_t^i)$ 为第 i 种蜜罐的部署成本。

优化过程采用“DQN 策略优化+Stackelberg 博弈”的方式实现，DQN 学习时序状态下最优 $\boldsymbol{D}_t$，模型确保 $\boldsymbol{D}_t$ 满足攻击均衡，实时调整蜜罐类型的部署策略，以适应不断变化的攻击类型。

2 算法设计

本文针对多类型蜜罐部署优化问题，提出了一种结合 Stackelberg 博弈与强化学习的动态博弈模型。

2.1 DQN 算法框架

DQN 通过神经网络近似 Q 函数，解决高维状态空间下的决策问题。本文 DQN 框架包含以下关键组件。

2.2 资源约束条件

攻防双方策略选择受总资源预算的严格限制：1) 防御资源约束 $\boldsymbol{C}_d^{\mathrm{T}}\boldsymbol{D}_t \leqslant \boldsymbol{R}_d$，$\boldsymbol{D}_t \in \Delta_n$；2)攻击资源约束 $\boldsymbol{C}_d^{\mathrm{T}}\boldsymbol{A}_t < \boldsymbol{R}_a$，$\boldsymbol{A}_t \in \Delta_m$。在 DQN 动作选择及博弈求解时，需对不满足上述资源约束条件的解进行惩罚或投影处理。

2.3 诱捕效能矩阵 $\boldsymbol{M}$

蜜罐诱捕能力如表 7 所示（如拟态蜜罐 0.96），攻击易感性为表 6 中“威胁程度×(1−攻击成本)”（如 SQL 注入攻击的攻击易感性为 0.9×(1−0.2)）。最终矩阵 $\boldsymbol{M}$ 为

$$\boldsymbol{M}=\begin{bmatrix} 0.59\times0.72 & 0.59\times0.42 & \cdots & 0.59\times0.21 \\ 0.75\times0.72 & 0.75\times0.42 & \cdots & 0.75\times0.21 \\ 0.82\times0.72 & 0.82\times0.42 & \cdots & 0.82\times0.21 \\ 0.96\times0.72 & 0.96\times0.42 & \cdots & 0.96\times0.21 \end{bmatrix} \tag{11}$$

2.4 策略优化目标与梯度

攻击者响应为

$$\boldsymbol{A}^*(\boldsymbol{D})=\underset{\boldsymbol{A}\in\Delta^m,\boldsymbol{C}_a^{\mathrm{T}}\boldsymbol{A}\leqslant\boldsymbol{R}_a}{\arg\max}\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{R}-\boldsymbol{L}^{\mathrm{T}}\boldsymbol{D})-\boldsymbol{C}_a^{\mathrm{T}}\boldsymbol{A} \tag{12}$$

防御者最优策略为

$$\boldsymbol{D}^*=\underset{\boldsymbol{D}\in\Delta^m,\boldsymbol{C}_a^{\mathrm{T}}\boldsymbol{A}\leqslant\boldsymbol{R}_d}{\arg\max}\boldsymbol{D}^{\mathrm{T}}\boldsymbol{M}\boldsymbol{A}^*(\boldsymbol{D})-\boldsymbol{C}_d^{\mathrm{T}}\boldsymbol{D} \tag{13}$$

在实际攻防环境中，攻击行为具有明显的时间序列特性，防御者根据历史趋势进行动态部署，将更有效地进行资源调度与诱捕。设计强化学习中的策略函数 π_θ 为带时序特征建模的神经网络。状态定义：设攻击者 t 时刻的行为向量 $\boldsymbol{A}_t \in R^m$，定义时间窗口大小为 k，定义时刻 t 的环境状态为攻击行为序列为 $\boldsymbol{S}_t=\{\boldsymbol{A}_{t-k},\boldsymbol{A}_{t-k+1},\cdots,\boldsymbol{A}_{t-1}\}$。此外，防御者的历史部署序列为 $\{\boldsymbol{D}_{t-k},\boldsymbol{D}_{t-k+1},\cdots,\boldsymbol{D}_{t-1}\}$，攻击收益变化率 $\Delta U_a=U_a(\boldsymbol{D}_{t-1},\boldsymbol{A}_{t-1})-U_a(\boldsymbol{D}_{t-2},\boldsymbol{A}_{t-2})$。其中，$\boldsymbol{A}_{t-i}$ 为攻击者在历史第 $t-i$ 步的攻击策略，$\boldsymbol{D}_{t-i}$ 为防御者对应时间 $t-i$ 的部署策略，U_{t-i}^a 为对应时间 $t-i$ 的攻击者收益。

状态嵌入向量 $\boldsymbol{h}(t)\in R^d$ 作为策略网络 π_θ 的输入，策略网络生成部署策略为

$$\boldsymbol{D}_t=\pi_\theta(h_t)\in\Delta^n,\boldsymbol{A}_t=\pi_\varnothing(D_t)\in\Delta^m \tag{14}$$

即时奖励为

$$r_t^d=U_d(\boldsymbol{D}_t,\boldsymbol{A}_t),r_t^a=U_a(\boldsymbol{D}_t,\boldsymbol{A}_t) \tag{15}$$

2.4.1 优势函数

引入折扣因子 $\gamma\in[0,1]$，用于控制奖励的衰减程度，定义防御者的优势函数为

$$A_t^{\text{adv}}=r_t^d+\gamma V(\boldsymbol{h}_{t+1})-V(\boldsymbol{h}_t) \tag{16}$$

其中，$V(h_t)$为状态值函数，表示策略网络在状态h_t下的长期回报，γ为折扣因子，值越接近1，表示奖励的重视程度越高，A_t^{adv}为当前策略相对于平均值函数的改进方向。

攻击者的优势函数为

$$A_t^{a\_\text{adv}} = r_t^a + \gamma V^a(\boldsymbol{D}_{t+1}) - V^a(\boldsymbol{D}_t) \tag{17}$$

2.4.2 目标函数与梯度

根据策略梯度定理[19]，策略的优化目标为最大化其在各个状态下的期望累计优势函数。防御者策略优化目标函数为

$$J_d(\theta) = E_t\left[\lg\pi_\theta(\boldsymbol{D}_t|\boldsymbol{h}_t)A_t^{\text{adv}}\right] \tag{18}$$

其中，E_t为对“时序攻击状态序列”的数学期望（遍历不同时间步t的攻击场景），$\lg\pi_\theta(\boldsymbol{D}_t|\boldsymbol{h}_t)$为防御者策略网络的对数概率（在状态$h_t$下选择部署策略$D_t$的概率对数），$A_t^{\text{adv}}$为防御者优势函数。

该目标函数的梯度表达式为

$$\nabla_\theta J_d(\theta) = E_t\left[\nabla_\theta\lg\pi_\theta(\boldsymbol{D}_t|\boldsymbol{h}_t)A_t^{\text{adv}}\right] \tag{19}$$

式(19)表示在状态h_t下，若某策略带来比预期更高的奖励，则该策略的选择概率π_θ被提升。

攻击者的策略优化目标函数及梯度定义为

$$J_a(\varnothing) = E_t\left[\lg\varnothing\pi_\theta(\boldsymbol{A}_t|\boldsymbol{D}_t)A_t^{a\_\text{adv}}\right] \tag{20}$$

其中，$\lg\pi_\varnothing(\boldsymbol{A}_t|\boldsymbol{D}_t)$为攻击者策略网络的对数概率（在防御者部署策略$\boldsymbol{D}_t$下选择攻击策略$\boldsymbol{A}_t$的概率对数），$A_t^{a_\text{adv}}$为攻击者优势函数，表示攻击者当前策略相对于“平均收益水平”的改进幅度。

$$\nabla_\varnothing J_a(\varnothing) = E_t\left[\nabla_\varnothing\lg\pi_\varnothing(\boldsymbol{A}_t|\boldsymbol{D}_t)A_t^{a\_\text{adv}}\right] \tag{21}$$

防御者与攻击者分别通过策略网络π_θ和$\pi_\varnothing$不断优化自身在博弈过程中的效用。随着训练的推进，若防御者策略未能有效适应攻击策略的增强，其效用将持续下降，而攻击者则可能逐步掌握主导权。

3 攻防主导角色动态切换机制

在时序环境中，攻防效用随策略迭代失衡，当攻击者效用连续上升且防御者效用下降时，系统自动调整领导者和跟随者身份，即切换为Stackelberg博弈中的主导方。

3.1 角色切换触发条件

定义“效用变化差”为量化攻防效用的动态趋势，触发条件如下。

防御者效用变化为

$$\Delta U_d = U_d(\boldsymbol{D}_t,\boldsymbol{A}_t) - U_d(\boldsymbol{D}_{t-1},\boldsymbol{A}_{t-1}) \tag{22}$$

攻击者效用变化为

$$\Delta U_a = U_a(\boldsymbol{D}_t,\boldsymbol{A}_t) - U_a(\boldsymbol{D}_{t-1},\boldsymbol{A}_{t-1}) \tag{23}$$

灵敏度阈值为

$$\varepsilon d = \mu\Delta U_d + \lambda\sigma\Delta U_d \tag{24}$$

其中，$\mu\Delta U_d$为防御者效用变化的均值，$\sigma\Delta U_d$为防御者效用变化的方差。

同理，攻击灵敏度为

$$\varepsilon a = \mu\Delta U_a + \lambda\sigma\Delta U_a \tag{25}$$

其中，λ为灵敏度因子，且$\lambda\in[0.5,1.5]$。

当满足以下条件时触发角色切换：1) 防御主导→攻击主导，$\Delta U_d < -\varepsilon d$，$\Delta U_a > \varepsilon a$（防御效用持续下降，攻击效用持续上升）；2) 攻击主导→防御主导，$\Delta U_d > \varepsilon d$，$\Delta U_a < -\varepsilon a$（防御效用上升，攻击效用下降）。

3.2 模型更改

对存在时序序列的动态攻击行为进行分析后，在式(1)的基础上，重新定义角色互换后的效用函数与策略网络输入，以适应新的主导关系。

3.2.1 效用函数更新

防御者效用函数更新为

$$U_d^{\text{new}} = \boldsymbol{D}_t^{\mathrm{T}}\boldsymbol{M}A_{t+1}^{\text{pred}} - \boldsymbol{C}_d^{\mathrm{T}}\boldsymbol{D}_t - \alpha\|\boldsymbol{D}_t - \boldsymbol{D}_{t-1}\|^2 \tag{26}$$

攻击者效用函数更新为

$$U_a^{\text{new}} = \boldsymbol{A}_t^{\mathrm{T}}(\boldsymbol{R} - \lambda\boldsymbol{M}^{\mathrm{T}}\boldsymbol{D}_t^{\text{pred}}) - \boldsymbol{C}_a^{\mathrm{T}}\boldsymbol{A}_t - \beta\|\boldsymbol{A}_t - \boldsymbol{A}_{t-1}\|^2 \tag{27}$$

其中，D_t^{pred}是对防御策略的历史加权平均，$\beta = 0.1$（攻击策略波动惩罚项），$\lambda = 0.8$（防御预测权重）。即便攻防角色发生互换（优化次序变为$\max_{\boldsymbol{A}}\max_{\boldsymbol{D}}$），由于新的效用函数依然保留了二次正则项，定理1中关于严格凹性和均衡唯一性的证明逻辑依然成立，这从理论上保证了系统在动态切换主导权的过程中，策略迭代依然能够收敛至新的纳什均衡点，避免了博弈过程中的振荡与发散。

3.2.2 策略网络适配

角色切换后，策略网络输入和输出需同步调整。策略网络的输入为

$$\boldsymbol{S}_t \in R^{k(m+n)} \tag{28}$$

式(28)包含过去k轮攻击分布$\boldsymbol{A}$和蜜罐部署策略$\boldsymbol{D}$。

策略网络的输出为

$$\boldsymbol{D}_t \in R^n \tag{29}$$

式(29)对应当前时刻蜜罐的部署策略。

该设计根据时序状态和对自身效用函数的优化，能够提高模型的自适应能力，有助于增强强化学习训练过程中策略的稳定性，提升蜜罐部署策略对复杂动态攻击环境的适应能力。

4　实验结果与分析

4.1　环境设置

本文搭建了一个中等规模的实验平台，包含13台物理服务器、2台低交互蜜罐、2台中交互蜜罐、2台高交互蜜罐、2台拟态蜜罐、2台真实主机、2台攻击机和1台管理机（管理蜜罐），多类型蜜罐部署方案如图1所示。

为验证多类型蜜罐部署方案在实际网络环境下的部署性能与攻防对抗效果，各类服务器具体配置与分工如表4所示。针对不同类型蜜罐具备的能力如表5所示。

8台蜜罐服务器根据攻防博弈给出的策略方案

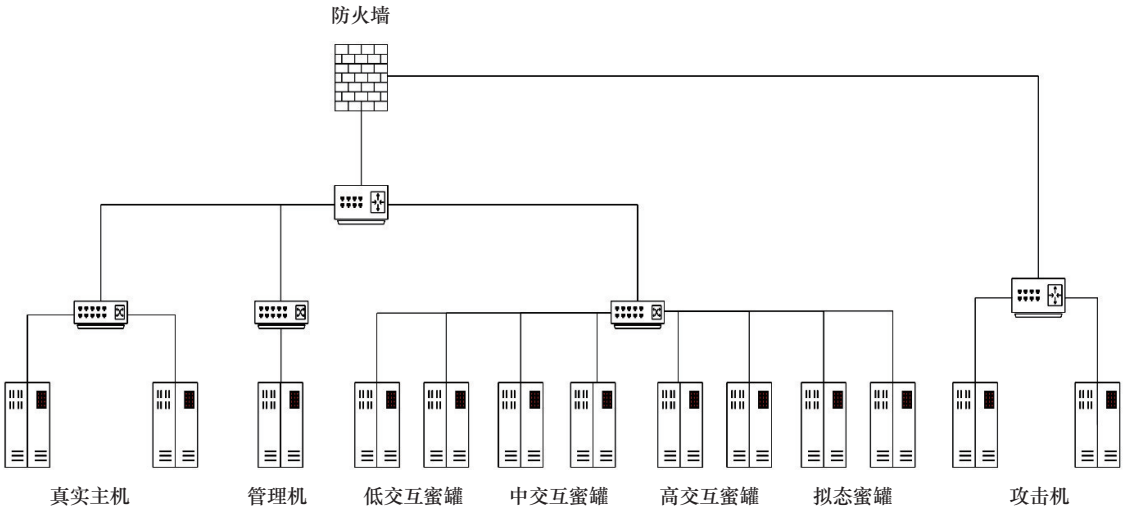

图1　多类型蜜罐部署方案

表4　实验主机指标

主机类型	台数/台	操作系统	漏洞数量/个	描述
真实主机	2	Centos7、Ubuntu	—	静态服务部署，运行于独立物理机、高隔离容器环境中
攻击机	2	Kali	—	具备攻击工具集合，可远程批量执行脚本攻击
管理机	1	Ubuntu	—	集中管理蜜罐主机、监控分析和联动防御
低交互蜜罐	2	Alpine linux	1	启动快速，适合大规模分布式部署
中交互蜜罐	2	Debian 10	2	使用脚本伪装响应逻辑，适度引诱攻击者
高交互蜜罐	2	Centos7	4	系统存在真实漏洞，需严格隔离，防止反向利用
拟态蜜罐	2	Centos7	6	镜像基于预定义模板自动生成，可动态更换指纹与服务组合，部署灵活，抗识别性强

表5　各类型蜜罐能力

蜜罐类型	蜜罐能力
低交互蜜罐	模拟基础协议响应，资源开销低但识别率较高
中交互蜜罐	部署部分真实组件与伪装逻辑，用于平衡资源与引诱能力
高交互蜜罐	部署完整服务，允许攻击者深度交互，具备良好的行为捕获能力
拟态蜜罐	采用Docker容器化部署技术，结合中间件特征异构和指纹伪装机制，提升诱骗性和抗识别性

进行动态部署，在高风险时段可以通过部署更多拟态蜜罐来提升对攻击者的诱捕效果，在稳定时期可以使用高交互和中交互蜜罐相结合，在保障诱骗性的前提下可以节省系统资源。

图1是本文实验环境的网络拓扑结构，该网络拓扑显示了攻击者的主机、所部署的蜜罐群组和真实服务器之间的网络拓扑分布情况。在拓扑图中，攻击流量可指向任意IP，系统通过策略调度模块来决定攻击流量是进入真实服务或者被重定向到对应的蜜罐容器，实现灵活的“主动诱捕”和“动态防御”机制。

在网络环境中，不同的攻击手段在威胁程度、攻击成本和潜在影响3个方面存在差异，本文选取Web攻击、SSH和FTP暴力破解及DoS攻击作为本次实验的攻击类型，Web攻击分别为SQL注入、跨站脚本攻击、服务器端请求伪造、文件包含漏洞、命令注入和服务器端模板注入。对每类攻击行为构建二维的参数向量，如表6所示，其中包括：1) 威胁程度，用于衡量攻击成功后对系统造成的破坏性，值域为[0,1]，数值越高代表攻击后果越严重；2) 攻击成本，指攻击者成功实施该类攻击所需的技术能力、工具复杂度与执行门槛。

从表6的数据来看，命令注入和SQL注入攻击对系统有较高的威胁，更易获取目标主机的资源，对系统服务造成危害。XSS和暴力破解虽然攻击成本低，但对目标系统服务造成的威胁较小，通过对攻击与防御成本的关联分析结合先验知识进行定量建模，为后续攻防博弈策略选择奠定基础。

各类型蜜罐部署成本及相关能力指标参数配置如表7所示。

表7　　蜜罐类型及相关指标

蜜罐类型	部署成本	运维复杂度	诱捕能力	被识别风险
低交互蜜罐	0.2	0.2	0.59	0.8
中交互蜜罐	0.5	0.6	0.75	0.5
高交互蜜罐	0.8	0.8	0.82	0.2
拟态蜜罐	0.9	0.9	0.96	0.1

4.2　拟态蜜罐部署方案

在本文实验中，分别部署两种形态的拟态蜜罐，并利用拟态蜜罐的异构中间件指纹和响应特征，提高对攻击者的诱捕能力。这两种蜜罐都运行在Docker容器中，功能分别是诱捕脚本攻击和SQL注入攻击。其仿真环境及相关服务如表8所示。

表8　　拟态蜜罐相关服务

平台类型	功能
WordPress博客平台	诱捕漏洞扫描和漏洞利用脚本攻击
PHPCMS平台	诱捕SQL注入攻击、命令注入攻击和文件包含漏洞攻击

传统蜜罐部署主要针对静态配置和单一攻击类型，难以适应动态环境，为了实现精准诱捕，提升蜜罐的诱捕效果，本文部署的拟态蜜罐旨在实现以下3个目标。

1) 伪装性：拟态蜜罐可以模拟真实的Web服务，从而提高其诱骗性。

2) 异构性：拟态蜜罐可以改变Web服务的中间件结构。

3) 可控性：拟态蜜罐可以被统一调度和进行

表6　　攻击类型及威胁程度

攻击类型	威胁程度	攻击成本	综合描述
SQL注入	0.9	0.2	工具成熟，且可直接操控数据库，影响范围广，后果严重
命令注入	0.95	0.60	直接执行系统命令，几乎必然导致服务器被攻陷，对攻击者技术要求较高
服务器端模板注入	0.8	0.55	模板注入可导致RCE，因模板引擎差异大，需针对性构造复杂payload
文件包含漏洞	0.75	0.45	可读取或执行服务器文件，严重时可链式触发远程代码执行
服务器端请求伪造	0.7	0.4	诱使服务器访问内网或受限资源，进而发起更深层次的攻击
跨站脚本攻击（XSS）	0.5	0.15	可窃取用户会话、篡改页面或执行恶意脚本，主要危害用户端
DoS攻击	0.6	0.7	通过大量无效请求占用服务器资源，导致合法用户无法正常访问服务
暴力破解	0.4	0.3	尝试所有可能的用户名和密码组合，获取账户访问权限

性能监控。

此外，为了在有限资源下实现蜜罐部署，实现资源利用最大化，拟态蜜罐的部署应遵循以下原则。

1) 轻量容器化：采用 Docker 容器化部署技术实现模块化部署，提升部署效率与迁移灵活性。

2) 行为一致性：通过镜像预设功能逻辑与页面结构，增强蜜罐与真实主机之间的相似性。

3) 定时拟态更新：依托构建流水线和脚本模板，定期更新蜜罐内部配置，避免长期暴露相同特征。

4) 统一调度和性能监控：可以和攻击诱捕调度模块相结合，实现拟态蜜罐的部署和蜜罐容器的运行状态采集。

拟态蜜罐运行在 Docker 容器中，部署流程包含蜜罐镜像构建、蜜罐容器运行和指纹跳变与拟态轮换 3 个阶段。

1) 蜜罐镜像构建。每类拟态蜜罐均对应一个基础镜像，预装所需服务组件与伪造数据集。系统采用自动构建脚本调用 Dockerfile 构建镜像。最终，镜像通过版本号标识区分不同拟态状态，并推送至本地私有镜像仓库中。

2) 蜜罐容器运行。构建完成后，通过统一的部署控制器调用 Docker API 启动容器，分别监听随机端口或由反向代理统一转发。每个容器运行后会自动注册至日志系统与调度中心，并开启自监控模块进行健康上报。

3) 指纹跳变与拟态轮换。系统根据既定时间窗口或触发事件，从策略仓库中获取新的拟态配置，自动触发镜像更新与容器重启流程。通过定时“跳变”页面结构、响应行为与资源路径，有效增加攻击者识别难度。

两种蜜罐在指纹层面相互补充，分别捕获不同攻击工具与脚本对各自目标环境的探测与利用行为。因此，本文方案针对多类型攻击可针对性部署蜜罐，能够对攻击进行有效诱捕，且智能决策模块可实时调整蜜罐部署方案，有效减少资源浪费，进一步体现了本文方案的有效性与创新性。

4.3　实验结果

针对攻击数据集，采用不同类型的蜜罐对攻击行为进行诱捕，在经过多次迭代后，其诱捕能力如图 2 所示，各类型蜜罐在 20~25 次迭代后曲线平缓收敛，低交互蜜罐约 0.61、中交互蜜罐约 0.75、高交互蜜罐约 0.82、拟态蜜罐约 0.96，由此凸显拟态蜜罐的策略跃升效果。

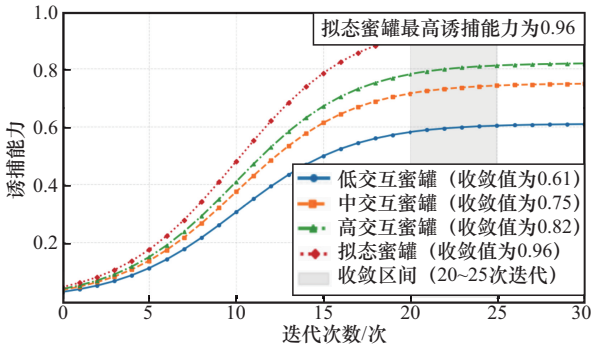

图 2　多类型蜜罐诱捕能力

攻防双方的效用函数变化曲线如图 3 所示。

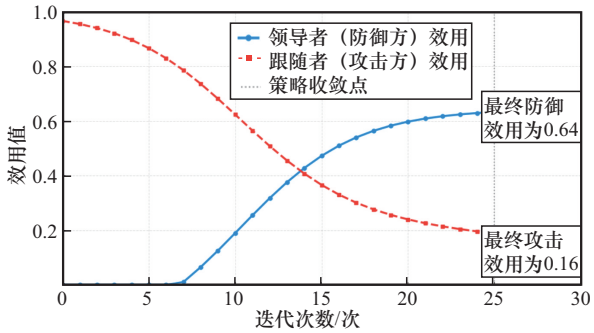

图 3　攻防双方的效用函数变化曲线

根据不同阶段的攻击策略选择，防御方通过强化学习后给出最优动态部署方案。Stage 0~Stage 4（每阶段约 15 min）4 类蜜罐部署策略比例随部署阶段演化的堆叠柱状图如图 4 所示。

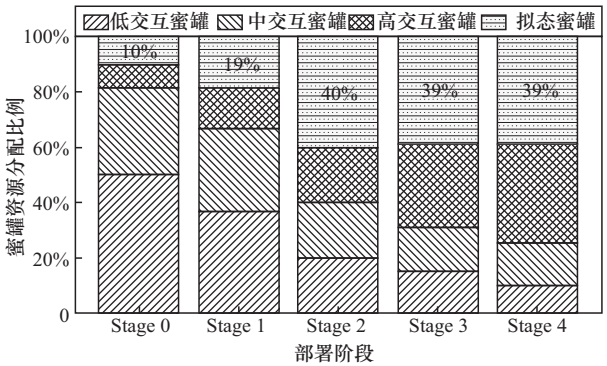

图 4　多阶段蜜罐部署策略

Stage 0：低交互蜜罐 50%、中交互蜜罐 31%、高交互蜜罐 9%和拟态蜜罐 10%。

Stage 1：低交互蜜罐 37%、中交互蜜罐 30%、

高交互蜜罐 14%、拟态蜜罐 19%。

Stage 2：低交互蜜罐降至 20%，拟态蜜罐跃升至 40%。

Stage 3~Stage 4：高交互蜜罐维持在 30% 以上，拟态蜜罐稳居 39%。

4.4　消融实验

为了验证各方案的有效性，本节设计了消融实验，比较了完整模型与以下变体，如图 5~图 7 所示。

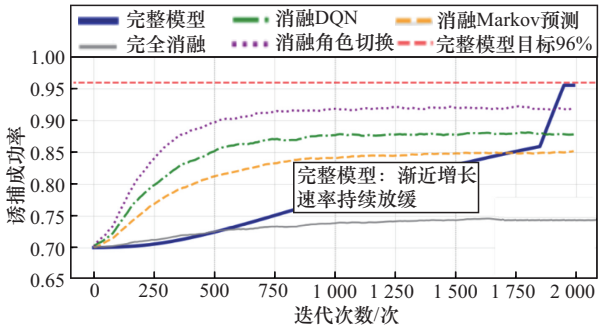

图 5　各方案诱捕成功率对比

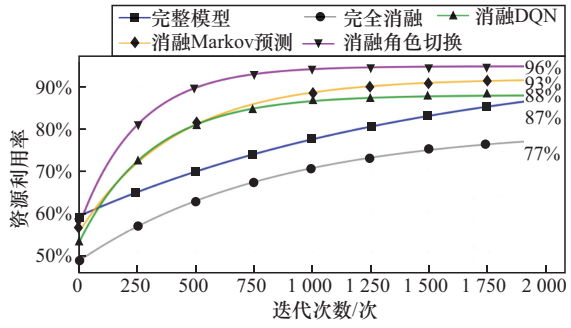

图 6　各方案资源利用率对比

1)消融 Markov 预测：使用随机预测代替 Markov 模型进行攻击预测。

2) 消融 DQN：使用随机策略进行蜜罐部署。

3) 消融角色切换：固定防御者为领导者，不进行角色切换。

4) 完全消融（单一蜜罐类型）：只使用拟态蜜罐进行部署。

通过 4 种消融实验对比，本文方案在诱捕成功率上优于其他几种方案。同时，由于本文方案在 SDN 架构上实现了多类型蜜罐智能部署，资源利用率处于平均水平。但整体来看，本文方案能够在有限资源下给出最优蜜罐部署方案，实现攻击诱捕。

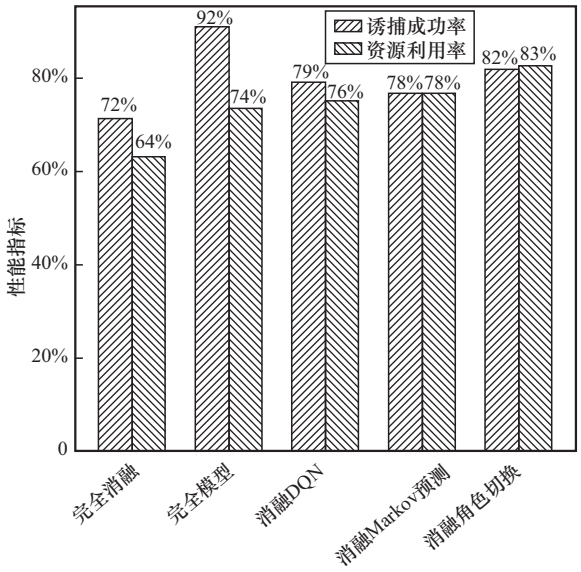

图 7　各方案诱捕成功率与资源利用率对比

4.5　相关工作对比

基于 CIC-IDS-2017 攻击数据集，表 9 给出了不同方案的性能对比。

1) 诱捕成功率：文献[7]采用单一高交互蜜罐架构，实现对 IoT 设备扫描、暴力破解等攻击的精准诱捕，诱捕成功率高达 82%；文献[12]部署单一 SDN 蜜罐，引流策略依赖静态规则配置，对未知攻击的识别和引流响应不及时，诱捕成功率较低；文献[11]采用 3 类 SSH 专用蜜罐阵列，诱捕成功率高于文献[7]和文献[12]，核心优势是采用多类型蜜罐和动态策略的组合，但因为该方法聚焦 SSH 协议攻击，无法响应其他攻击类型，导致诱捕成功率未突破 90%；本文采用多类型蜜罐组合，构建攻击类型与蜜罐类型的动态匹配模型，最终诱捕成功率

表 9　不同方案的性能对比

方案	诱捕成功率	策略更新时延/s	资源利用率	蜜罐种类	是否适应多类型攻击
文献[7]	82%	12	65%	单一高交互	否（仅 IoT）
文献[12]	78%	8	60%	单一 SDN	否（单一类型）
文献[11]	85%	15	70%	3 类	否（仅 SSH）
本文方案	96%	5	92%	4 类	是（8 类攻击）

高达96%。

2) 策略更新时延：文献[7]采用传统硬件部署方式，策略更新需手动配置或本地服务器下发指令，导致策略更新时延较长；文献[11]采用博弈理论优化策略，需要基于历史攻击数据进行多轮迭代计算，增加耗时，导致其策略更新时延最长；文献[12]SDN架构的集中式控制器为策略更新提供了便利，但是策略更新需要依赖预设规则触发，缺乏实时攻击的动态决策机制，更新触发及时性不足；本文所采用的集中式决策+分布式执行架构，通过自动化策略调度和优化机制，实现攻击特征识别–策略优化–蜜罐适配的快速闭环，能够及时响应新型攻击，提升方案的动态适配能力。

3) 资源利用率：文献[12]的资源利用率最低，采用单一SDN蜜罐部署，蜜罐功能单一，导致大量网络资源浪费；文献[7]采用单一高交互IoT蜜罐，提升了诱捕效果，但占用较多计算资源用于模拟设备运行与交互，导致资源利用率不高；文献[11]使用多类型蜜罐，能够充分利用网络资源，但仅限于SSH协议攻击的设计，导致资源利用率低于80%；本文采用多类型蜜罐协同部署，将资源有效分配给各类型蜜罐进行协同防御，有效避免了资源闲置，实现了资源利用率的最大化。

4) 蜜罐种类和多类型攻击的适应性：文献[7]仅部署一类IoT专用高交互蜜罐，聚焦于IoT设备安全防御，不适应多类型攻击；文献[12]部署单一SDN蜜罐，核心优势是通过SDN控制器下发规则，实现对攻击流量的有效引流，不适应多类型攻击；文献[11]聚焦于SSH协议攻击，能覆盖SSH相关攻击行为，但对SSH协议之外的攻击类型存在明显的局限性；本文通过分析CIC-IDS-2017攻击数据集的多类型攻击特征，部署多类型蜜罐组合，突破了单一协议或者单一攻击类型的限制，通过动态策略调度机制，让不同类型蜜罐针对性适配相对应的攻击，提升了方案的通用性和实用性。

综上，本文方案与文献[7]、文献[11]和文献[12]的核心差异在于“多类型蜜罐部署”与“基于CIC-IDS-2017攻击数据集的攻击特征精准适配”。其他方案均存在蜜罐类型单一、攻击适配范围窄、策略更新不及时或资源利用率低的问题，且均无法适配多类型攻击。而本文方案通过多类型蜜罐协同部署、精细化资源调度与快速策略更新机制，不仅实现了96%的高诱捕成功率，还在资源受限的情况下将资源利用率提升至92%。对比传统静态部署策略下的平均防御效用约为0.47，而本文方案收敛至0.64，提升了约35%。同时可适配多类型攻击，显著优于现有方案，有效解决了单一蜜罐方案的场景局限与资源浪费问题，为复杂网络环境中的多类型攻击防御提供了更优解。

5 结束语

针对传统静态蜜罐部署方案在复杂网络环境中存在蜜罐类型单一、动态适应性不足的问题，本文以CIC-IDS-2017攻击数据集合为支撑，构建了Stackelberg博弈与DQN算法相融合的多类型蜜罐部署模型，实现固定资源约束下低交互、中交互、高交互和拟态蜜罐的最优动态调度。一方面，考虑不同类型蜜罐的部署成本差异化与诱捕能力不同，平衡防御效能与资源开销；另一方面，结合攻击行为的时序演化特征，通过Stackelberg博弈中攻防主导角色的动态切换与DQN策略优化，实现部署策略的动态调整，有效提升了蜜罐防御对攻击的自适应性。

本文目前在多类型蜜罐调度和攻击诱捕方面已取得了良好的进展，但针对未知攻击的诱捕仍有待提高。在未来研究中，本文将结合多源场景融合和大模型未知检测两个方面进行改进：1) 聚焦多源融合攻击场景，构建跨域多类型蜜罐协同防御机制，突破单域防御局限；2) 引入大语言模型增强攻击意图深度预测能力，进一步提升部署方案的智能化与精准性。

参考文献：

[1] Ahmed M, Mahmood A N, Hu J K. A survey of network anomaly detection techniques[J]. Journal of Network and Computer Applications, 2016, 60: 19-31.

[2] Mell P M, Grance T. The NIST definition of cloud computing[J]. Communications of the ACM, 2011, 53(6), 50-55.

[3] García S, Grill M, Stiborek J, et al. An empirical comparison of botnet detection methods[J]. Computers & Security, 2014, 45: 100-123.

[4] Bhuyan M H, Bhattacharyya D K, Kalita J K. Network anomaly detection: a survey and comparative analysis of contemporary solutions[J]. Computers & Security, 2014, 45, 100-123.

[5] Spitzner L. Honeypots: tracking hackers[M]. Boston: Addison-Wesley Publishing Company, 2002.

[6] Qin X S, Jiang F, Cen M C, et al. Hybrid cyber defense strategies using Honey-X: a survey[J]. Computer Networks, 2023, 230: 109776.

[7] Yang X Y, Yuan J, Yang H, et al. A highly interactive honeypot-based approach to network threat management[J]. Future Internet, 2023, 15(4): 127.

[8] Piggin R, Buffey I. Active defence using an operational technology honeypot[C]//Proceedings of the 11th International Conference on System Safety and Cyber-Security (SSCS). Institution of Engineering and Technology, 2016: 1-6.

[9] 邬江兴. 网络空间拟态防御研究[J]. 信息安全学报, 2016, 1(4): 1-10.
Wu J X. Research on cyber mimic defense[J]. Journal of Cyber Security, 2016, 1(4): 1-10.

[10] Guan C Q, Liu H T, Cao G H, et al. HoneyIoT: adaptive high-interaction honeypot for IoT devices through reinforcement learning[C]// Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks. New York: ACM Press, 2023: 49-59.

[11] Shi L Y, Wang X R, Hou H W. Research on optimization of array honeypot defense strategies based on evolutionary game theory[J]. Mathematics, 2021, 9(8): 805.

[12] 凌颖, 杨春燕, 黎新, 等. 一种基于遗传算法改进型的蜜罐防入侵方法和电子设备:CN202410526963.9[P]. 2024-04-29.
Ling Y, Yang C Y, Li X, et al. An improved honeypot auti-intrusion method and electronic device based on genetic algorithm: CN202410526963.9[P]. 2024-04-29.

[13] 王夕冉, 石乐义, 赵志豪, 等. 一种自适应差分进化动态蜜罐策略优化方法:CN202410249092.0[P]. 2024-03-05.
Wang X R, Shi L Y, Zhao Z H, et al. An adaptive differential evolution dynamic honeypot strategy optimization method: CN202410249092.0[P]. 2024-03-05.

[14] 宋丽华, 张津威, 张少勇. 基于博弈论对手建模的物联网SSH自适应蜜罐策略[J]. 信息网络安全, 2023, 23(11): 38-47.
Song L H, Zhang J W, Zhang S Y. An adaptive IoT SSH honeypot strategy based on game theory opponent modeling[J]. Netinfo Security, 2023, 23(11): 38-47.

[15] 王鹃, 杨泓远, 樊成阳. 一种基于多阶段攻击响应的SDN动态蜜罐[J]. 信息网络安全, 2021, 21(1): 27-40.
Wang J, Yang H Y, Fan C Y. A SDN dynamic honeypot with multi-phase attack response[J]. Netinfo Security, 2021, 21(1): 27-40.

[16] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540): 529-533.

[17] Liu Y, Li Z, Chen X, et al. Dynamic honeypot deployment based on stackelberg game in SDN[J]. In IEEE INFOCOM, 2022.

[18] Sutton R S, Barto A G. Reinforcement Learning[M]. Cambridge: MIT Press, 2018.

[19] Daskalakis C, Foster D J, Golowich N. Independent policy gradient methods for competitive reinforcement learning[J]. Advances in Neural Information Processing Systems, 2020, 33: 5527-5540.

[作者简介]

韩雨（1996– ），男，陕西商洛人，西安理工大学博士生，主要研究方向为蜜罐诱捕、攻防博弈。

陈元恒（2002– ），男，河南新乡人，郑州大学硕士生，主要研究方向为网络攻防、蜜罐诱捕。

王一川（1983– ），男，河南开封人，博士，西安理工大学教授、博士生导师，主要研究方向为系统脆弱性分析、网络攻防对抗、隐匿网络等。

马艺宾（1999– ），男，河南洛阳人，西安理工大学硕士生，主要研究方向为网络安全态势感知。

黑新宏（1976– ），男，陕西延安人，博士，西安理工大学教授、博士生导师，主要研究方向为人工智能、网络安全、轨道交通安全等。