导读：工业互联网作为数字化转型的关键基础设施，其安全边界正从封闭的生产控制网络向开放、互联、异构的复杂生态系统扩展。传统“单点防御、静态配置”的安全模式难以应对高级持续性威胁与供应链级联风险。本专栏围绕“主动检测-动态加固-精准防护-系统治理”的技术链条，遴选出四篇具有代表性的研究成果，分别从智能渗透测试、协议动态自适应加固、控制场景协同防护以及供应链系统性治理四个层面，尝试构建更加弹性、协同的工业互联网安全体系。期望本专栏能够为工业互联网安全领域的学术研究与行业实践提供有益借鉴。

专栏主编：

洪晟，北京航空航天大学副教授，博士生导师，从事信息网络安全、人工智能与大数据、智能制造/工业互联网等方向的教学和科研工作。北京市安全学科带头人，中国工程院和中国国际经济交流中心战略规划专家。兼任工信部工业数据空间技术组副组长、中国电子节能技术协会数据安全专委会副主任、中国机电一体化协会专委会副秘书长等职。主持和参与973/863课题、国家重点研发课题、国家自然科学基金等课题30余项；担任国内外多个期刊和会议的编委，发表论文100余篇，其中SCI检索30余篇，全球前1% ESI高被引论文3篇；授权国家发明专利30余项，获国防科技进步一等奖1项，北京市科技进步二等奖1项。

基于DPPO的双智能体协同渗透测试方法研究

李　科，霍朝宾，贺敏超，杨　继
（华北计算机系统工程研究所，北京　100083）

摘　要：基于强化学习的自动化网络渗透测试方法近年来受到广泛关注，但现有研究多局限于中小规模网络环境，难以应对实际网络中主机数量庞大、拓扑复杂的挑战。提出一种基于DPPO（Dual-agent Proximal Policy Optimization）的双智能体协同渗透测试框架，通过分工协作机制提升大规模网络环境下的攻击决策效率。该框架包含目标决策智能体（TargetDecider）与攻击执行智能体（AttackExecutor），分别负责目标节点筛选与具体漏洞利用执行，二者通过共享环境状态进行协同，以协作奖励机制引导，从而实现分工合作。实验在基于CyberBattleSim构建的仿真网络环境中进行，并与传统的DQN、PPO、SAC等智能体方法进行了比较，结果表明DPPO框架攻陷关键目标所需的步骤数更少、波动更小，显示出更高的攻击效率与策略稳定性。同时，其敏感主机攻陷比率维持在较高水平，表明该框架能可靠完成核心渗透任务。在累计奖励方面，DPPO亦表现出持续优化与稳健收敛的趋势。实验结果表明，该框架能够有效适应大规模静态仿真网络中的渗透测试任务。该框架对动态防御和真实网络环境的适用性仍需在后续研究中进一步验证。

关键词：自动化渗透测试；双智能体协作；深度强化学习
中图分类号：TP393.08　　**文献标志码：**A　　**DOI：**10.19358/j.issn.2097-1788.2026.06.001

中文引用格式：李科，霍朝宾，贺敏超，等. 基于DPPO的双智能体协同渗透测试方法研究[J]. 网络安全与数据治理，2026，45(6)：1-8.
英文引用格式：Li Ke，Huo Chaobin，He Minchao，et al. Research on dual-agent collaborative penetration testing method based on DPPO[J]. Cyber Security and Data Governance，2026，45(6)：1-8.

Research on dual-agent collaborative penetration testing method based on DPPO

Li Ke，Huo Chaobin，He Minchao，Yang Ji
（National Computer System Engineering Research Institute of China，Beijing 100083，China）

Abstract：The automated network penetration testing method based on reinforcement learning has received widespread attention in recent

years. However, the existing research is mostly limited to small and medium-sized network environments, making it difficult to address the challenges posed by the vast number of hosts and complex topology in real-world networks. This study proposes a dual-agent collaborative penetration testing framework based on DPPO (Dual-agent Proximal Policy Optimization), which enhances the efficiency of attack decision-making in large-scale network environments through a division of labor and collaboration mechanism. The framework comprises a target decision-making agent (TargetDecider) and an attack execution agent (AttackExecutor), responsible for target node selection and specific vulnerability exploitation execution, respectively. The two agents collaborate through shared environmental states, guided by a collaborative reward mechanism, to achieve division of labor and cooperation. Experiments were conducted in a simulated network environment built based on CyberBattleSim, and compared with traditional agent methods such as DQN, PPO, and SAC. The results show that the DPPO framework requires fewer steps and exhibits less fluctuation to compromise key targets, demonstrating higher attack efficiency and strategic stability. At the same time, its sensitive host compromise rate remains at a high level, indicating that the framework can reliably complete core penetration tasks. In terms of cumulative reward, DPPO also shows a trend of continuous optimization and robust convergence. The results indicate that the framework can effectively adapt to penetration testing tasks in large-scale static simulated networks, while its applicability to dynamic defense scenarios and real network environments still requires further validation.

Key words: automated penetration testing; dual-agent collaboration; deep reinforcement learning

0 引言

随着网络攻击手段日趋复杂，传统静态防御体系已难以有效应对新型威胁。自动化渗透测试作为主动安全评估的重要手段，能够通过模拟攻击行为提前发现系统脆弱性，但其传统实现模式高度依赖专家经验，存在成本高、适应性差等问题[1]。近年来，强化学习技术因其在序列决策问题上的优势，被引入自动化渗透测试领域，通过将网络环境建模为马尔可夫决策过程[2]，使智能体能够自主学习攻击策略。然而，现有方法在面对大规模复杂网络时仍面临显著挑战：状态与动作空间随网络规模指数增长导致探索效率低下，以及渗透测试任务中奖励信号稀疏导致的收敛困难。

针对上述问题，本文提出一种“目标决策—攻击执行”双智能体协同训练框架。该框架将渗透测试任务分解为宏观目标选择和微观攻击执行两个层次，分别由目标决策智能体（TargetDecider）和攻击执行智能体（AttackExecutor）承担，二者通过状态共享与协同奖励机制实现高效分工协作。本文主要贡献包括：（1）设计了面向大规模网络的双智能体分层强化学习架构，有效降低了学习复杂度；（2）提出了与分工机制相匹配的协同训练方法及分层奖励函数；（3）在 CyberBattleSim 仿真环境中验证了方法的有效性，实验表明本框架在攻击效率与稳定性方面优于传统单智能体方法。

1 相关研究

当前，自动化渗透测试研究主要围绕以下三类方法展开。首先，基于强化学习的方法通过智能体与环境的持续交互来学习攻击策略，已成为主流研究方向[3-4]。例如，Zhang 等人[5]提出改进深度循环 Q 网络以处理部分可观测环境下的渗透测试决策问题，Zhou 等人[6]改进了深度 Q 网络以提升策略稳定性，Tran 等人[7]和曾庆伟等人[8]采用分层强化学习方法进行攻击路径发现与任务分解，Nguyen 等人[9]和 Tran 等人[10]则尝试通过双智能体分工及级联强化学习来处理复杂决策问题。此外，近期研究也开始关注渗透测试策略的可解释性以及情节记忆机制的应用[11-12]。其次，基于图论、启发式搜索和知识图谱的方法利用多启发式信息融合或知识图谱对网络进行建模，并借助启发式算法规划攻击路径[13-14]。再次，基于生成对抗模仿学习的方法能够生成多样化的测试用例序列[15]，然而其性能受训练数据质量限制，在应对未知漏洞和实际执行转化方面仍存在明显约束。总体来看，现有方法在应对大规模、动态变化的真实网络场景时，普遍面临适应性不足、协同效率有限等挑战。

与已有双智能体或多智能体渗透测试工作相比，本文 DPPO 框架的核心差异主要体现在三个方面：一是在协同机制上，已有双智能体方法多侧重于两个攻击智能体并行探索或采用 Actor-Critic 结构划分决策与评价，多智能体方法则通常将不同节点或攻击者视为相对独立的协作主体；本文将 TargetDecider 与 AttackExecutor 分别建模为“目标选择—攻击执行”的上下级协同关系，通过目标节点传递和共享环境状态形成显式分工。二是在任务解耦方式上，已有工作多从动作空间分层或多智能体并行搜索角度降低复杂度，而本文直接依据渗透测试流程将全局路径规划与局部漏洞利用解耦，使宏观目标优先级学习和微观攻击动作学习分别在更小子空间中完成。三是在奖励设计上，本文不是简单复用统一回报，而是为两个 PPO 智能体分别设计目标成功、域控制器攻陷、攻击动作有效性

和无效动作惩罚等分层奖励，使奖励信号与各自职责对应，从而强化协同而非仅追求个体收益最大化。

2 基于双智能体的强化学习模型

2.1 强化学习仿真环境

本研究基于开源仿真平台 CyberBattleSim 构建训练与测试环境。为贴近真实企业内网渗透场景，本文设计并实现了两个不同规模的网络环境：LargeNetwork-v0（约 15 个节点）与 HugeNetwork-v0（约 50 个节点）。具体设计考虑如下：

（1）网络拓扑与资产配置。环境模拟了包含多种设备类型的企业网络，包括工作站、文件服务器、Web 服务器、数据库服务器及域控制器等。不同类型的节点在开放服务、存在的漏洞类型、初始权限及防御配置等方面均存在差异。这种设计使得网络中存在明显的高价值目标（如域控制器），为智能体学习区分攻击优先级提供了必要场景。具体配置见表 1。

表 1　实验环境的具体配置

配置项	LargeNetwork-v0	HugeNetwork-v0
网络规模	中型企业网络	大型企业网络
工作站节点/个	6	30
域控制器/个	1	2
文件共享服务器/个	2	4
Web 服务器/个	2	4
数据库服务器/个	2	4
Linux 服务器/个	2	6
用户账户/个	20~99	800
本地漏洞/种	10	10
远程漏洞/种	5	5
服务端口/种	5	5

（2）部分可观测设置。为反映真实渗透中攻击者信息有限的情况，环境采用部分可观测设定。智能体初始仅知晓入口节点信息，网络拓扑、节点属性、可用凭证等均需通过扫描、漏洞利用等动作逐步发现。观察空间定义为当前已知的节点特权状态、已发现的节点列表、已掌握的凭证等信息构成的向量。

（3）动作空间与有效性约束。环境提供一组原子级攻击动作，包括本地漏洞利用、远程漏洞利用、凭证连接等。为降低无效探索，本文引入了动作掩码机制，在每个时间步动态屏蔽不符合当前状态的动作（例如，尝试攻击一个尚未被发现的节点）。此外，动作空间天然呈现“选择目标节点”与“执行具体攻击”两类操作，这与本文提出的双智能体分工设计在逻辑上是一致的。

2.2 双智能体强化学习框架

针对在复杂渗透任务中决策效率低下的问题，本文提出一种分工明确、协同训练的双智能体框架。该框架是一个由两个 PPO 智能体组成的协同系统。为更直观地体现这一“双 PPO 智能体”（Dual PPO agents）的协同设计，并与单一 PPO 智能体方案相区分，后文将其统称为双智能体 DPPO（Dual-agent PPO，DPPO）框架。

2.2.1 整体架构

本文将自动化渗透测试任务建模为部分可观测的马尔可夫决策过程（POMDP），由两个协同工作的智能体共同求解。该 POMDP 可形式化定义为八元组$\langle \mathcal{S}, \mathcal{O}^s, \mathcal{O}^e, \mathcal{A}^s, \mathcal{A}^e, \mathcal{R}^s, \mathcal{R}^e, \mathcal{P} \rangle$，其中 $\mathcal{S}$ 为环境状态空间，$\mathcal{O}^s$ 和 $\mathcal{O}^e$ 分别为 TargetDecider 与 AttackExecutor 的观察空间，$\mathcal{A}^s$ 和 $\mathcal{A}^e$ 为动作空间，$\mathcal{R}^s$ 和 $\mathcal{R}^e$ 为奖励函数，$\mathcal{P}$ 为状态转移概率。两个智能体以分层决策方式协作：在每个决策步 t，TargetDecider 根据当前观察 $\boldsymbol{o}_t^s \in \mathcal{O}^s$ 选择目标节点 g_t，随后 AttackExecutor 基于观察 $\boldsymbol{o}_t^e \in \mathcal{O}^e$ 和目标 g_t 执行具体攻击动作序列。两智能体策略分别表示为 $\pi^s(a_t^s \mid \boldsymbol{o}_t^s)$ 和 $\pi^e(a_t^e \mid \boldsymbol{o}_t^e, g_t)$，通过 PPO 算法独立训练，但通过协同奖励设计实现行为协调。

2.2.2 观察空间建模

观察空间设计为部分可观测，以模拟真实渗透测试中攻击者对网络信息的有限掌握。TargetDecider 的观察向量 $\boldsymbol{o}_t^s$ 由四部分构成：节点特权级别向量 $\boldsymbol{p}_t \in \mathbb{R}^N$，其中 $p_i>0$ 表示智能体已控制节点 i；已发现节点总数 $d_t \in \mathbb{N}$；当前掌握的凭证数量 $c_t \in \mathbb{N}$；以及动作掩码向量 $\boldsymbol{m}_t \in \{0,1\}^{|\mathcal{A}^s|}$，指示哪些动作在当前状态下有效。AttackExecutor 的观察向量 $\boldsymbol{o}_t^e$ 则包含节点特权级别向量 $\boldsymbol{p}_t$、当前节点服务与漏洞特征向量 $\boldsymbol{s}_t \in \mathbb{R}^K$、凭证数量 c_t 以及 TargetDecider 指定的目标节点 ID $g_t \in \mathbb{N}$。这种差异化的观察设计使 TargetDecider 关注网络宏观拓扑与资产价值，而 AttackExecutor 专注于当前节点的攻击面与目标节点信息。

2.2.3 动作空间设计

动作空间经过抽象设计，与 CyberBattleSim 仿真环境的原始动作空间保持对齐。TargetDecider 的动作空间 $\mathcal{A}^s$ 定义为对已发现但未控制节点的选择，即 $\mathcal{A}^s = \{a_k^s\}_{k=1}^{K_s}$，其中 $K_s = N_{\text{discovered}} - N_{\text{owned}}$。每个动作对应一个候选目标节点，智能体选择下一个要攻陷的目标。AttackExecutor 的动作空间 $\mathcal{A}^e = \{a_k^e\}_{k=1}^{K_e}$ 则包含三类原子攻击动作：本地漏洞利用 $a^e_{\text{local}}(v, \text{vid})$，在已控节点上执行本地权限提升或信息收集；远程漏洞利用 $a^e_{\text{remote}}(v_{\text{src}}, v_{\text{target}}, \text{vid})$，从已控节点攻击目标节点；以及连接攻击 $a^e_{\text{connect}}(v_{\text{src}}, v_{\text{target}}, \text{port}, \text{cred})$，使用掌握的凭证

尝试认证连接。通过 CyberBattleSim 的抽象动作层，这些离散动作索引被映射为具体的环境可执行动作。

2.2.4 奖励函数构建

奖励函数采用分层设计，为两个智能体提供差异化的学习信号。TargetDecider 的即时奖励 R_t^s 由探索奖励 R_{discover}、目标成功奖励 $R_{\text{targetsuccess}}$、域控制器奖励 R_{dc} 和步数惩罚 R_{step} 四部分加权组成，数学表示为：

$$R_t^s=\lambda_1 R_{\text{discover}}+\lambda_2 R_{\text{targetsuccess}}+\lambda_3 R_{\text{dc}}+\lambda_4 R_{\text{step}} \tag{1}$$

其中，探索奖励采用归一化形式，计算公式为：

$$R_{\text{discover}}=0.3\times(\Delta N_{\text{node}}/N_{\text{total}}+\Delta N_{\text{cred}}/C_{\max}) \tag{2}$$

其中，ΔN_{node} 和 ΔN_{cred} 为本步新增的节点数和凭证数，N_{total} 为网络总节点数，$C_{\max}$ 为最大凭证容量。以此鼓励智能体积极探测未知信息。目标成功奖励在 AttackExecutor 成功攻陷当前所选目标时给予固定值+5.0。域控制器奖励是核心导向信号，当智能体成功攻陷域控制器等高价值目标时，给予+100.0 的高额奖励，以明确最终任务。步数惩罚则为每步-0.1，旨在促使智能体规划更短的攻击路径，提升决策效率。

AttackExecutor 的即时奖励 R_t^e 则聚焦于攻击动作的有效性，由漏洞利用成功奖励 $R_{\text{exploitsuccess}}$、目标达成奖励 $R_{\text{goalreach}}$ 和无效动作惩罚 $R_{\text{invalidpenalty}}$ 组成，数学形式为：

$$R_t^e=\beta_1 R_{\text{exploitsuccess}}+\beta_2 R_{\text{goalreach}}+\beta_3 R_{\text{invalidpenalty}} \tag{3}$$

漏洞利用成功奖励根据攻击类型差异化设定：本地漏洞利用为+1.0，远程漏洞利用为+2.0，凭证连接攻击为+1.5，以体现不同攻击手段的难度与价值。目标达成奖励在成功攻陷 TargetDecider 指定的目标时给予+10.0 的基础奖励；若该目标为域控制器，则额外增加+100.0，与 TargetDecider 的域控制器奖励形成双重强化。无效动作惩罚对于不满足前置条件的攻击尝试，根据违规程度给予-0.1～-0.2 的惩罚，引导智能体学习攻击动作的逻辑约束。

上述奖励函数的各项系数 λ_i 和 β_i 均通过网格搜索实验调优确定。调优过程中采用交叉验证，以确保正向奖励足以引导期望行为，同时负向惩罚能够有效抑制无效探索。最终确定的系数组合为：$\lambda=[1.0,5.0,100.0,-0.1]$；攻击类型奖励系数 $\beta_{\text{type}}=[1.0,2.0,1.5]$（对应本地、远程、连接三类攻击），目标达成奖励系数 $\beta_{\text{goal}}=10.0$，无效动作惩罚系数 $\beta_{\text{invalid}}=-0.1$。该设计使两个智能体在分工基础上形成隐式协作，共同推动渗透任务的高效完成。

2.2.5 策略网络与训练算法

两个智能体均采用基于 Actor-Critic 架构的 PPO 算法进行训练。每个智能体包含两个神经网络：Actor 网络 $\pi_\theta(a\mid o)$ 输出动作概率分布，Critic 网络 $V_\varphi(o)$ 估计状态价值。PPO 的目标函数通过限制策略更新幅度确保训练稳定性：

$$L^{\text{CLIP}}(\theta)=\mathbb{E}_t[\min(r_t(\theta)\hat{A}_t,\text{clip}(r_t(\theta),1-\epsilon,1+\epsilon)\hat{A}_t)] \tag{4}$$

其中，$r_t(\theta)=\pi_\theta(a_t\mid o_t)/\pi_{\theta_{\text{old}}}(a_t\mid o_t)$ 为新旧策略的概率比，$\hat{A}_t=G_t-V_\varphi(o_t)$ 为优势函数（G_t 为折扣累积回报），ϵ 为裁剪超参数（本文取 0.2）。Critic 网络的更新则通过最小化均方误差实现：

$$L(\varphi)=\mathbb{E}_t[(V_\varphi(o_t)-G_t)^2] \tag{5}$$

在网络结构的设计上，TargetDecider 与 AttackExecutor 均采用多层感知机（MLP），但根据各自任务特点设置了不同的隐藏层维度。TargetDecider 需要综合全局信息进行长远规划，因此其网络容量较大：Actor 网络包含 3 层全连接，隐藏层神经元数量分别为 256 和 512，输入维度为观察空间 $\boldsymbol{o}_t^s$ 的维度（由节点特权向量、已发现节点数、凭证数及动作掩码组成），输出维度为动态变化的可选目标数；Critic 网络结构与之类似，输出为标量状态价值。AttackExecutor 聚焦于当前攻击面的微观细节，网络规模相对较小以加快训练：Actor 网络同样为 3 层全连接，隐藏层神经元数量分别为 128 和 256，输入维度为观察空间 $\boldsymbol{o}_t^e$ 的维度（包括节点特权向量、当前节点特征、凭证数及目标节点 ID），输出维度为固定的原子动作数；Critic 网络则对应输出状态价值。所有隐藏层均采用 ReLU 激活函数，Actor 网络的输出层使用 Softmax 函数生成动作概率，Critic 网络输出层则为线性激活。表 2 给出了具体的双智能体策略网络结构参数。

表 2　双智能体策略网络结构参数

智能体	网络组件	层数	神经元数量	输入维度	输出维度
Target-Decider	Actor	3	256→512→K_s	$\boldsymbol{o}_t^s$	可选目标数（动态）
	Critic	3	256→512→1	同上	1
Attack-Executor	Actor	3	256→512→K_e	$\boldsymbol{o}_t^e$	原子动作数（固定）
	Critic	3	256→512→1	同上	1

上述差异化设计的根本原因在于：TargetDecider 需从宏观角度评估全网节点的价值与可达性，其决策依赖于对大量全局信息的融合，因此需要更大的网络容量来捕捉复杂的依赖关系；而 AttackExecutor 的任务是在给定目标后执行具体的攻击序列，主要关注当前节点与目标节点的局部特征，较小规模的网络足以拟

合其策略，同时能减少计算开销、提升训练效率。这种设计使两个智能体在各自擅长的领域发挥优势，共同完成复杂的渗透测试任务。

2.2.6 协同训练流程

图 1 描述了双智能体协同训练的整体架构。在训练过程中，TargetDecider 与 AttackExecutor 采用独立 PPO 范式进行学习，即各自维护独立的策略网络与经验回放缓冲区，并基于自身收集的交互数据分别进行策略更新。两智能体之间不共享梯度或参数，其协同行为主要通过环境状态传递与分层奖励设计实现。每个训练回合中，TargetDecider 首先选择目标节点，随后 AttackExecutor 在最大步数 $K_{\max}$ 内尝试攻陷该目标。经验回放缓冲区独立维护，定期采样批次数据更新网络参数。整个训练过程持续进行，直到达到预设回合数或成功率达到收敛阈值。该框架通过任务分解降低了单智能体的学习复杂度，通过协同奖励促进了两智能体的行为协同，最终实现了在复杂网络环境中高效、自适应的自动化渗透测试策略学习。

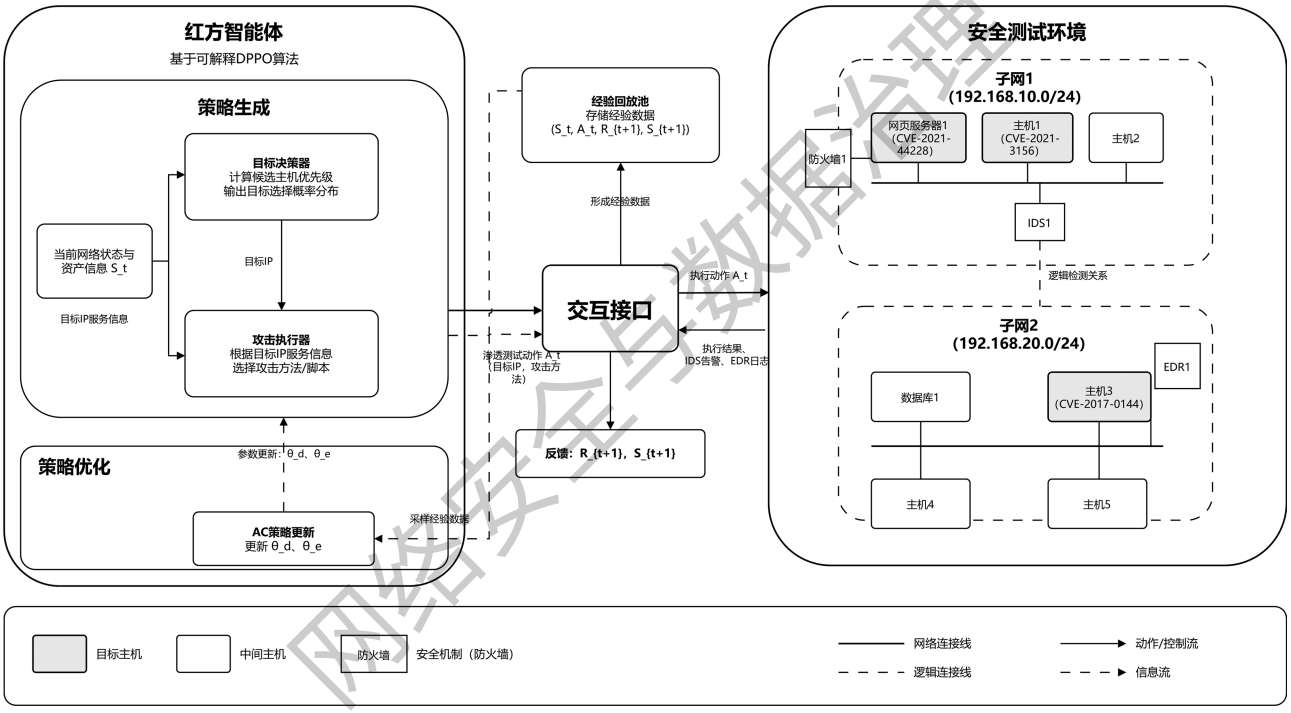

图 1　双智能体协同训练的整体架构

3　实验设计与结果分析

本节旨在通过系统的对比实验，评估所提出的“结构探索—漏洞利用”双智能体分层强化学习框架 DPPO 的性能。实验围绕两个核心问题展开：(1)在复杂的大规模网络环境中，DPPO 相较于主流单智能体强化学习基线（DQN、PPO、SAC）在攻击效率、成功率与收敛速度上是否具有优势；(2)该框架在不同规模网络（LargeNetwork-v0 与 HugeNetwork-v0）中的泛化与适应性如何。

3.1　实验设置

3.1.1　环境与参数

实验基于第 2.1 节所述的两个定制化 CyberBattleSim 仿真环境：

（1）LargeNetwork-v0：中型企业网络，包含约 15 个节点，用于验证模型在适度复杂度下的基础性能。

（2）HugeNetwork-v0：大规模企业网络，包含约 50 个节点（工作站、服务器、域控制器、数据库服务器等），用于评估模型在高维状态与动作空间下的扩展性与鲁棒性。

所有对比实验均采用相同的随机种子初始化，以保证公平性。每个算法的训练总回合数（episode）设定为 1 000，每回合最大步数为 1 000。关键超参数（学习率、折扣因子、经验回放缓冲区大小）均经过网格搜索调优，以确保各基线算法处于其最佳配置。

3.1.2　评估指标

为全面衡量算法性能，本文采用以下四个核心指标。(1)累计奖励：每个训练回合中获得的总奖励，反映算法策略的整体收益与探索效率。(2)收敛轮次：

算法达到其最大平滑奖励 90%所需的最少训练回合数，用于量化学习速度与稳定性。(3)敏感主机攻陷比率：在训练过程中，成功攻陷高价值目标（如域控制器、数据库服务器）的回合占总回合的比例，衡量战略目标达成能力。(4)域控制器攻陷步骤：在成功攻陷域控制器的回合中，所花费的环境步数，直接反映攻击路径规划的效率与精准度。

3.1.3 对比基线

为验证 DPPO 框架的有效性，选取了三种广泛使用的、具有代表性的深度强化学习算法作为基线。(1) DQN：基于值函数的经典算法，使用深度网络拟合 Q 值，通过 ε-贪婪策略探索。(2) PPO：基于策略梯度的主流算法，以训练稳定、采样效率高著称，作为单智能体策略优化的强基线。(3) SAC：基于最大熵原理的先进算法，在连续控制任务中表现卓越，其离散版本被用于本实验。

所有基线模型均使用与 DPPO 中 TargetDecider 同等的观察空间，并采用相同的环境交互接口，确保对比的公平性。

此外，为避免因奖励设计差异造成评价偏差，三类基线算法在训练与评价时均采用与 DPPO 一致的环境奖励函数和终端目标判定，即节点发现、凭证获取、漏洞利用成功、域控制器攻陷及无效动作惩罚等奖励/惩罚项保持一致；DPPO 仅在内部将这些奖励按 TargetDecider 与 AttackExecutor 的职责分配为分层信号，最终曲线统计仍使用同一评价口径。

3.2 LargeNetwork-v0 环境下的性能分析

图 2 和图 3 展示了各算法在攻击效率与综合收益两个维度的表现差异。在“攻陷域控制器所需步数”这一关键指标上，DPPO 表现最优：其初始步数为 90~100 步，训练后稳定在 300 步左右，显著低于其他算法。PPO 算法的步数从 150 步逐步上升至 500 步以上，路径冗余明显；DQN 步数维持在 450 步左右，但后期波动剧烈，稳定性不足。这表明 DPPO 通过目标选择与攻击执行的分工机制，能够学习到更直接、稳定的攻击路径，快速达成核心目标。

在累计奖励方面，各算法呈现相反趋势。PPO 累计奖励快速上升并稳定在 600 以上，说明其善于获取节点发现等中间奖励；DPPO 累计奖励稳定在 160 左右，增长平缓；SAC 则几乎失效。这一反差表明 DPPO 具有明确的“目标导向”特征，优先追求终端目标而非过程奖励，而 PPO 等单智能体虽积累较多过程奖励，却导致攻击路径冗余。

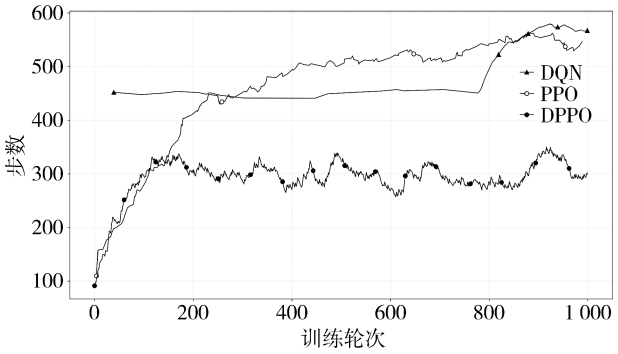

图 2　LargeNetwork-v0 环境下成功攻陷域控制器所需步骤数

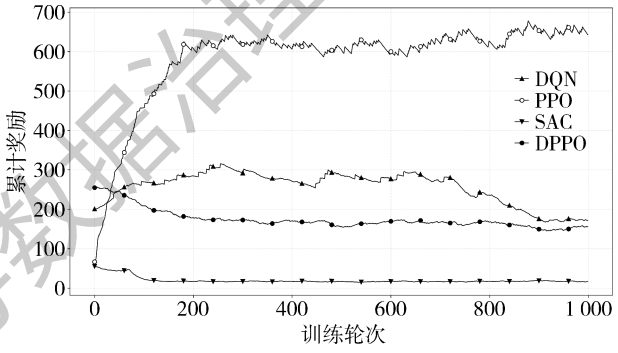

图 3　LargeNetwork-v0 环境下的累计奖励

需要说明的是，图 3 中的累计奖励是在上述奖励函数与评价口径一致条件下得到的结果。因此，DPPO 累计奖励较低并不意味着任务完成能力较弱，而是说明其策略较少通过反复扫描、扩展节点等过程行为累积中间奖励，更多地将有限交互步数用于接近并攻陷域控制器。换言之，在奖励函数保持一致的前提下，累计奖励与攻击效率之间存在一定张力：PPO 等基线可能通过更多中间动作获得较高过程奖励，但同时引入路径冗余；DPPO 虽然牺牲部分过程奖励，却以更少步骤达成终端目标。

在真实渗透场景中，攻陷核心目标的步骤数比累计奖励更具实际意义：更少的步骤意味着更短的暴露时间和更低的检测概率。因此，尽管 DPPO 在累计奖励上不占优，但其在攻击效率与稳定性上的优势验证了双智能体分工架构的有效性。

在收敛速度方面，DPPO 也优势明显：达到最大平滑奖励 90%所需的收敛轮次约为 320 回合，而 PPO 约 580 回合，DQN 约 720 回合，SAC 未能收敛。这一效率提升源于任务解耦对学习复杂度的降低。单智能体需在“目标选择×攻击方式”的联合空间中学习，动作空间规模大（如 LargeNetwork 中约为 150），导致样本效率低；而 DPPO 将问题分解为 TargetDecider

（目标选择，复杂度 O（N））和 AttackExecutor（局部攻击，动作空间固定），每个智能体在更小子空间探索，从而提升样本效率并加速价值估计收敛。

带来的学习效率提升体现在两个方面。（1）缩小探索空间。每个智能体只需在其专属的子空间中探索，避免了在大量无关动作上浪费样本，从而提高了样本效率。（2）加速价值估计收敛。Critic 网络只需估计子空间内的状态价值，拟合难度降低，优势函数估计更准确，从而加快策略更新速度。

3.3 HugeNetwork-v0 环境下的性能分析

HugeNetwork-v0 环境规模大、节点类型多、攻击路径复杂，对算法的探索能力与长期规划能力提出了极高要求。

从图 4 与图 5 的实验结果可以看出，在该复杂环境下，各算法的性能表现呈现出显著差异。本文提出的 DPPO 框架在累计奖励方面呈现稳定上升的趋势，约在 400 个训练回合后进入高奖励区间，且曲线波动较小，表现出较好的学习稳定性。在反映算法实际攻击效能的敏感主机攻陷比例这一指标上，DPPO 能够稳定维持在约 80%的水平，表明该框架不仅能够获得较高的累积回报，更重要的是能够可靠、可重复地完成对高价值目标的渗透任务。

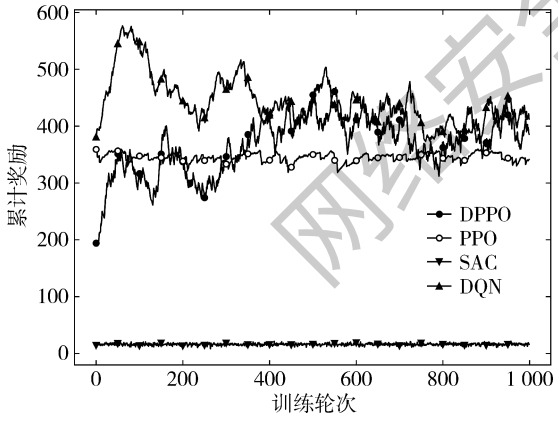

图 4　HugeNetwork-v0 环境下累计奖励曲线

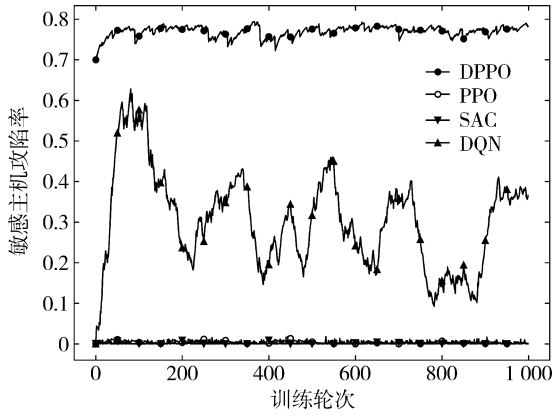

图 5　HugeNetwork-v0 环境下敏感主机攻陷比率变化曲线

相比之下，传统算法在这一复杂环境中暴露出明显不足。DQN 虽然在部分训练回合中能够获取较高的奖励，但其奖励曲线与攻陷比例均表现出较大幅度的波动，策略稳定性较差，显示出其在复杂探索与利用平衡问题上的局限性。PPO 算法则出现早熟收敛现象，其奖励曲线快速趋于平稳并维持在中等水平，敏感主机攻陷比例接近为零，表明其保守的更新机制限制了算法在复杂环境中的有效探索能力，使其难以突破局部最优。SAC 算法在该环境下表现较差，未能形成有效的攻击策略。

3.4　综合分析

综合以上实验结果，可以得出以下结论。

从收敛速度与稳定性来看，DPPO 框架较各单智能体基线表现出明显优势。该框架通过将全局渗透任务分解为结构探索与漏洞利用两个层次，降低了单智能体的决策复杂度，从而在一定程度上缓解了稀疏奖励与大规模动作空间带来的探索困难。在最终策略性能方面，DPPO 在累计奖励与敏感主机攻陷比例两项指标上均优于对比算法，表明其学习到的联合策略不仅具备较高的攻击效率，也能够稳定达成预设的渗透目标。

从攻击过程的效率与鲁棒性来看，DPPO 在攻陷关键目标时所需步骤更少，不同回合间的表现波动也更小，说明该框架所习得的策略具有较好的可复现性与泛化能力，而非依赖随机探索的偶然结果。此外，各基线算法在复杂环境中也表现出一定的局限性：SAC 在大规模网络中虽能获得一定奖励，但攻击路径效率不高；PPO 训练过程较为稳定，但策略收敛速度较慢；DQN 则始终难以平衡探索与利用，策略波动显著。这些现象进一步表明，针对渗透测试任务中动作逻辑依赖强、奖励信号稀疏等特点，有必要进行面向任务特性的算法设计与优化。

本研究通过在不同规模仿真环境中的对比实验，验证了双智能体分层强化学习框架在应对复杂网络渗透测试任务时的有效性，表明其在攻击效率、目标达成率与策略稳定性等多个关键维度上均体现出一定优势，为后续面向实际场景的自动化渗透测试技术研究提供了参考。

同时需要指出，本文实验仍局限于 CyberBattleSim 构建的静态仿真环境，网络拓扑、漏洞集合与服务配

置均为预设项，尚未覆盖真实网络中常见的动态防御、入侵检测告警反馈、未知漏洞出现、漏洞修补以及攻击—防御对抗等变化。后续应在隔离靶场或数字孪生网络中引入动态主机状态、IDS/IPS 响应、随机漏洞暴露与防御策略切换，并与真实渗透测试流程中的人工路径规划或现有自动化工具进行对比，以进一步验证其工程实用性和鲁棒性。

4 结论

本文针对自动化渗透测试任务中因状态空间复杂、奖励稀疏导致的决策效率瓶颈，提出了一种基于目标选择与漏洞利用分工的双智能体协同强化学习框架。通过将渗透任务分解为目标选择与攻击执行两个子问题（分别由 TargetDecider 与 AttackExecutor 承担），并设计相应的奖励机制，有效降低了学习复杂度，提升了策略的收敛效率与可解释性。但是，当前 DPPO 框架仍存在若干固有局限。首先，双智能体独立更新可能带来目标不一致风险：TargetDecider 选择的目标在宏观上具有较高价值，但 AttackExecutor 在局部攻击面受限或凭证不足时可能难以执行，导致协同效率下降。对此，可在后续工作中引入目标可达性估计、共享价值函数或集中训练—分散执行机制，使目标选择同时考虑局部可执行性。其次，本文环境中的漏洞集合为预设已知漏洞，模型对零日漏洞、未知服务指纹和动态漏洞修补的适应性仍有限，可结合在线探索、元学习或漏洞知识图谱更新机制提升迁移能力。再次，双 PPO 智能体需要维护两套策略网络和经验缓存，在大规模网络或高频动态环境下会带来较高训练开销，对此，可通过参数共享、轻量化策略网络、优先经验采样与并行仿真加速降低计算成本。基于上述不足，未来研究将重点面向动态攻防靶场开展验证，并探索仿真到真实网络的安全迁移与在线微调方法。

参考文献

[1] Chen Yang, He Junjiang, Fang Wenbo, et al. Automatic penetration testing model based on reinforcement learning for complex network environments [J]. The Journal of Supercomputing, 2025, 81(11): 1169.

[2] Li Qianyu, Zhang Min, Shen Yi, et al. A hierarchical deep reinforcement learning model with expert prior knowledge for intelligent penetration testing [J]. Computers & Security, 2023, 132(12): 103358.

[3] Ghanem M C, Chen T M. Reinforcement learning for efficient network penetration testing[J]. Information, 2020, 11(1): 6.

[4] Hu Z, Beuran R, Tan Y. Automated penetration testing using deep reinforcement learning[C]// 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW). IEEE, 2020: 2-10.

[5] Zhang Yue, Liu Jingju, Zhou Shicheng, et al. Improved deep recurrent Q-network of POMDPs for automated penetration testing[J]. Applied Sciences, 2022, 12(20): 10339.

[6] Zhou Shicheng, Liu Jingju, Hou Dongdong, et al. Autonomous penetration testing based on improved deep Q-network[J]. Applied Sciences, 2021, 11(19): 8823.

[7] Tran K, Akella A, Standen M, et al. Deep hierarchical reinforcement agents for automated penetration testing[J]. arXiv preprint arXiv: 2109. 06449, 2021.

[8] 曾庆伟，张国敏，邢长友，等. 基于分层强化学习的智能化攻击路径发现方法[J]. 计算机科学，2023，50(7)：308-316.

[9] Nguyen H, Teerakanok S, Inomata A, et al. The proposal of double agent architecture using actor-critic algorithm for penetration testing[C]// Proceedings of the 7th International Conference on Information Systems Security and Privacy (ICISSP), 2021: 440-449.

[10] Tran K, Standen M, Kim J, et al. Cascaded reinforcement learning agents for large action spaces in autonomous penetration testing[J]. Applied Sciences, 2022, 12(21): 11265.

[11] Goel D, Moore K, Wang J, et al. Unveiling the black box: a multilayer framework for explaining reinforcement learning-based cyber agents[J]. arXiv: 2505. 11708, 2025.

[12] Zhou Ziqiao, Zhou Tianyang, Xu Jinghao, et al. Efficient penetration testing path planning based on reinforcement learning with episodic memory[J]. Computer Modeling in Engineering & Sciences, 2024, 140(3): 2613-2634.

[13] 胡泰然，臧艺超，曹蓉蓉，等. 基于多启发式信息融合的攻击路径发现算法研究[J]. 信息安全学报，2021，6(3)：202-211.

[14] Liang Rufeng, Chen Junhan, Chen Xingchi, et al. Application research of knowledge graph in automated penetration testing path planning in the digital era[C]//Computational and Experimental Simulations in Engineering: ICCES 2024. Cham: Springer, 2025: 321-330.

[15] Chen Jinyin, Hu Shulong, Zheng Haibin, et al. GAIL-PT: an intelligent penetration testing framework with generative adversarial imitation learning[J]. Computers & Security, 2023, 126: 103055.

（收稿日期：2026-02-02）

作者简介：

李科（1999-），男，硕士研究生，主要研究方向：网络安全。

霍朝宾（1983-），男，硕士，正高级工程师，主要研究方向：信息安全。

贺敏超（1989-），男，硕士，工程师，主要研究方向：工业控制系统信息安全技术。

版权声明

凡《网络安全与数据治理》录用的文章，如作者没有关于汇编权、翻译权、印刷权及电子版的复制权、信息网络传播权与发行权等版权的特殊声明，即视该文章署名作者同意将该文章的汇编权、翻译权、印刷权及电子版的复制权、信息网络传播权与发行权授予本刊，本刊有权授权本刊合作数据库、合作媒体等合作伙伴使用。同时，本刊支付稿酬已包含上述使用的费用，特此声明。

《网络安全与数据治理》编辑部