面向智能渗透攻击的欺骗防御方法

陈晋音[1,2]，胡书隆[1,2]，邢长友[3]，张国敏[3]
（1. 浙江工业大学信息工程学院，浙江 杭州 310023；2. 浙江工业大学网络空间安全研究院，浙江 杭州 310023；
3. 陆军工程大学指挥控制工程学院，江苏 南京 210007）

摘 要：基于强化学习的智能渗透攻击旨在将渗透过程建模为马尔可夫决策过程，以不断试错的方式训练攻击者进行渗透路径寻优，从而使攻击者具有较强的攻击能力。为了防止智能渗透攻击被恶意利用，提出一种面向基于强化学习的智能渗透攻击的欺骗防御方法。首先，获取攻击者在构建渗透攻击模型时的必要信息（状态、动作、奖励）；其次，分别通过状态维度置反扰乱动作生成，通过奖励值符号翻转进行混淆欺骗，实现对应于渗透攻击的前期、中期及末期的欺骗防御；最后，在同一网络环境中展开 3 个阶段的防御对比实验。实验结果表明，所提方法可以有效降低基于强化学习的智能渗透攻击成功率，其中，扰乱攻击者动作生成的欺骗方法在干扰比例为 20%时，渗透攻击成功率降低为 0。

关键词：强化学习；智能渗透攻击；攻击路径；欺骗防御
中图分类号：TP393.08
文献标志码：A
DOI: 10.11959/j.issn.1000−436x.2022202

Deception defense method against intelligent penetration attack

CHEN Jinyin[1,2], HU Shulong[1,2], XING Changyou[3], ZHANG Guomin[3]
1. College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China
2. Institute of Cyber Space Security, Zhejiang University of Technology, Hangzhou 310023, China
3. College of Command & Control Engineering, Army Engineering University, Nanjing 210007, China

Abstract: The intelligent penetration attack based on reinforcement learning aims to model the penetration process as a Markov decision process, and train the attacker to optimize the penetration path in a trial-and-error manner, so as to achieve strong attack performance. In order to prevent intelligent penetration attacks from being maliciously exploited, a deception defense method for intelligent penetration attack based on reinforcement learning was proposed. Firstly, obtaining the necessary information for the attacker to construct the penetration model, which included state, action and reward. Secondly, conducting deception defense against the attacker through inverting the state dimension, disrupting the action generation, and flipping the reward value sign, respectively, which corresponded to the early, middle and final stages of the penetration attack. At last, the three-stage defense comparison experiments were carried out in the same network environment. The results show that the proposed method can effectively reduce the success rate of intelligent penetration attacks based on reinforcement learning. Besides, the deception method that disrupts the action generation of the attacker can reduce the penetration attack success rate to 0 when the interference ratio is 20%.

Keywords: reinforcement learning, intelligent penetration attack, attack path, deception defense

收稿日期：2022−07−07；**修回日期**：2022−09−29
基金项目：国家自然科学基金资助项目（No.62072406）；浙江省重点研发计划基金资助项目（No.2021C01117）；2020 年工业互联网创新发展工程基金资助项目（No.TC200H01V）；浙江省万人计划科技创新领军人才基金资助项目（No.2020R52011）
Foundation Items: The National Natural Science Foundation of China (No.62072406), The Key Research and Development Program of Zhejiang Province (No.2021C01117), The 2020 Industrial Internet Innovation Development Project (No.TC200H01V), The Ten Thousand Talents Program of Zhejiang Province (No.2020R52011)

0　引言

随着互联网技术的不断发展与广泛应用，各种网络攻击行为带来诸多网络安全问题。网络渗透测试[1]是一种用于评估网络系统安全的有效方法，测试人员通过模拟黑客攻击行为对网络及其主机进行漏洞挖掘并展开安全评估[2]。然而，渗透测试的非法使用将成为威胁网络安全的一种攻击手段。因此，实施针对渗透攻击的防御至关重要。

传统的手动渗透攻击依赖于安全专家的经验与操作，自动化渗透攻击则将现有的漏洞扫描工具（如 Namp[3]、Nexpose[4]等）集成至同一渗透框架（如 Metasploit[5]、Core Impact[6]等）内，来执行半自动化的渗透攻击。智能渗透攻击将渗透攻击过程建模为马尔可夫决策过程，利用不同的算法模型训练攻击者以不断试错的方式对目标网络进行渗透，获得最佳渗透策略和最优渗透路径，从而实现高效渗透。现有对智能渗透攻击的研究主要包括基于传统强化学习（RL, reinforcement learning）[7]和基于深度强化学习[8]两类算法模型。例如，Zhou 等[9]与 Tran 等[10]分别提出了新的深度强化算法模型来实现智能渗透攻击，并以此提高其渗透性能；通过改进版深度 Q 网络算法（NDD-DQN, noisy double dueling-deep Q network）及分层深度强化学习（HRL, hierarchical deep reinforcement learning）来克服渗透攻击在大规模场景下因动作高维离散[11]和奖励稀疏导致渗透攻击过程难以稳定收敛的问题。相比于手动渗透攻击与自动渗透攻击，基于强化学习的渗透攻击具有更强的攻击性能，因此本文针对这类渗透攻击展开欺骗防御研究。

Yuill[12]和 Gartner[13]等提出了网络欺骗[14-15]的定义，网络欺骗防御的核心是防御方通过干扰和误导攻击者的认知决策过程，使其采取有利于防御方的动作，从而有助于防御方检测、延缓或中断攻击过程，实现增强目标网络安全性的目的。王硕等[16]根据多阶段渗透攻击的特点将其全过程分为渗透攻击初期、中期及末期 3 个阶段。在渗透攻击初期，攻击者利用扫描工具（如 Nmap）对目标网络进行扫描，获取主机的 IP 地址、端口号、操作系统版本号、潜在漏洞运行的服务等信息，并借助渗透工具（如 Metasploit）实现对主机的入侵。在渗透攻击初期，主要采取掩盖真实信息和模拟虚假节点的方式来达到欺骗防御的目的，如 Jafarian 等[17]设计的基于对手感知的地址随机化方法和 Wang 等[18]构造的随机域名与地址跳变（RDAM, random domain name and address mutation）防御方法均使攻击者扫描到错误的 IP 地址，进而使攻击者无法成功进行后续的攻击行为。在渗透攻击中期，攻击者已经深入目标网络内部，探测识别潜在的主机漏洞，进而渗透这些漏洞，并不断横向移动以逼近攻击目标。在目标网络内部部署蜜罐是抵抗渗透攻击中期攻击的主流方法，如 Anagnostakis 等[19]将入侵检测设备判定为异常的流量牵引至“影子蜜罐”来提高针对未知缓冲区溢出攻击的检测准确率。在渗透攻击末期，防御方的重点是实现对攻击目标的特殊防护，如 Rowe 等[20]和石乐义等[21]提出的伪蜜罐和拟态蜜罐警戒色都是为了使攻击者将真实系统当作蜜罐从而将攻击者吓退。

已有的渗透攻击欺骗防御研究主要针对非智能渗透攻击方法，而对于智能渗透攻击被恶意利用对目标网络所造成的安全威胁，目前尚无相关研究对其进行有效防御。基于此，本文针对基于强化学习的智能渗透攻击在训练过程中其经验数据存在被欺骗的可能性，如状态中的漏洞运行服务以及操作系统类型数据被修改等，提出一种面向智能渗透攻击的欺骗防御方法。本文的主要工作如下。

首先，将渗透攻击建模为马尔可夫决策过程并利用强化学习算法 Q 学习（QL, Q-learning）算法[22]对目标网络进行智能渗透。其次，通过对该渗透环境下的状态、动作以及奖励所对应的实际含义进行分析，修改状态、动作以及奖励训练数据中的关键信息，扰乱攻击者的渗透策略生成，使其攻击策略出错或失效，从而实现对攻击者的欺骗防御。最后，将修改攻击者状态、动作以及奖励数据的过程对应于多阶段渗透攻击的前期、中期和后期 3 个阶段，并在同一网络环境进行 3 个阶段的对比实验，以此验证所提方法的防御性能。

1　基于强化学习的智能渗透攻击

1.1　强化学习

强化学习是一种基于智能体与环境之间序列交互的机器学习方法，智能体在给定时间内通过不断试错学习以最大化长期累积奖励 G_t [8]。该过程通常用马尔可夫决策过程（MDP, Markov decision process）来描述，MDP 可以表示为一个四元组 $<S, A, R, P>$，其中，S 表示状态空间集合，A 表

示动作空间集合，R 表示奖励函数，P 表示状态转移矩阵，即从当前 t 时刻状态 s_t 采取动作 a_t 转移到下一个时刻状态的概率 $P(s_{t+1}\mid s_t,a_t)$。累积奖励表示为

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \quad (1)$$

其中，$\gamma \in (0,1)$ 表示折扣因子，用于衡量当前奖励对未来奖励的重要性。

QL[22]是一种基于 Q 值函数的强化学习算法，主要思想是将智能体不同时刻的状态和动作构建为存储 $Q(s,a)$ 值的 Q 表，然后根据 Q 值来选取能够获得最大收益的动作。Q 表的更新机制为

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a') - Q(s,a)] \quad (2)$$

其中，s 和 a 分别表示攻击者当前的状态和动作，s' 表示攻击者采取动作 a 后出现的下一个状态，a' 表示攻击者在状态 s' 下可能采取的动作，r 表示即时奖励，α 和 γ 分别表示学习率和折扣因子。

1.2 智能渗透攻击过程

强化学习在游戏领域的决策任务性能表现突出，例如 Alpha Go[22]、Dota2[23]和 StarCraft Ⅱ[24]，基于强化学习的攻击者已经超过了人类玩家。与许多 Gym 和 Atari[22]游戏规则一样，渗透攻击也是一个基于环境状态的动态决策过程。因此，可以训练基于强化学习的智能渗透攻击[26]智能体来观察动态网络环境并通过试错学习最优策略。

在基于 QL 算法实现智能化渗透攻击[27]的过程中，攻击者会不断优化自动生成的渗透攻击路径，并最终得到一条最佳攻击路径[28]。攻击者在训练过程中会根据不同主机运行的不同漏洞服务，如安全外壳（SSH, secure shell）协议、超文本传输协议（HTTP, hypertext transfer protocol）、文件传输协议（FTP, file transfer protocol）等，选择相应的具有代表性的漏洞来计算该漏洞利用的概率，本文根据通用漏洞评分系统（CVSS, common vulnerability scoring system）分别对 SSH 服务和 HTTP 服务设置 0.9 的渗透概率，对 FTP 服务设置 0.6 的渗透概率。除了渗透操作外，针对提权操作和扫描操作的概率均设置为 1。攻击者在采取对应的操作后，环境会反馈其相应的奖励值，该奖励值计算式如式(3)所示，主要包括主机的价值以及对该主机进行漏洞利用、提权或者扫描操作所消耗的成本。

$$R = \sum_{h \in H} \text{value}(h) - \sum_{a \in A} \text{cost}(a) \quad (3)$$

其中，H 表示网络中可以被攻击者渗透的主机集合，A 表示攻击者的动作集合。在训练过程中，攻击者的目标就是最大化累积奖励值，如式(4)所示，以此获取对目标敏感主机的渗透路径，即需要以尽可能少的操作来渗透最大价值的目标敏感主机。

$$\max_{\pi} E\left[\sum_{t=0}^{T} \gamma^t R(s_t,a_t,s_{t+1}) \mid \pi\right] \quad (4)$$

针对本文的蜜罐目标网络环境，目标敏感主机奖励值设为 100，蜜罐主机的奖励值设为−100，攻击者在网络中只能与相连的子网或者主机之间进行渗透，每个回合中攻击者的结束条件有 2 种情况：1) 获得所有目标敏感主机的 Root 权限；2) 回合训练步数达到了设置的最大值。

攻击者以不断试错的方式来获得最大化累积奖励值，从而学习到渗透目标敏感主机的最优渗透路径。在蜜罐网络中，由于蜜罐的奖励值为负值，因此攻击者在攻击过程中会绕过蜜罐主机，最终学习到利用较少的操作步骤来攻击目标敏感主机的最优渗透路径。

2 欺骗防御问题建模

将渗透攻击建模为马尔可夫决策的过程中，智能体被视为渗透攻击者，$S = \{s_1, s_2, \cdots, s_i\}$ 表示攻击者的状态集合，其中 s_i 是攻击者在某个特定时刻 i 利用扫描工具对目标网络进行扫描，能获取到关于主机的 IP 地址、端口号、操作系统版本号、潜在漏洞运行服务等信息，随后攻击者将进一步对目标主机进行渗透入侵。$A = \{a_1, a_2, \cdots, a_i\}$ 代表攻击者的动作集合，其中 a_i 是攻击者根据其与环境交互获得的状态 s_i 所采取的动作。在渗透攻击过程中，攻击者采取的动作主要包括漏洞扫描、漏洞利用以及权限提升等操作。最后，根据此刻攻击者采取的动作 a_i 是否成功渗透至目标主机，环境将给予其奖励反馈 r_i，$R = \{r_1, r_2, \cdots, r_n\}$ 表示攻击者获得的即时奖励集合。

定义 1 $S_d = \{s'_1, s'_2, \cdots, s'_i\}$ 为欺骗防御方的状态集合，即欺骗状态，其中 s'_i 对应于攻击方在 i 时刻被防御方欺骗的状态，用一个 x 维的布尔向量 $\boldsymbol{d}$ 表示。维度 x 由目标网络的子网数 n_{sub}、主机数 n_{host}、不同子网中最大主机数 $n_{\max}$ 以及网络配置信息类型数 n_{con} 确定。

$$x = (n_{\text{host}} + 1)(n_{\text{sub}} + n_{\max} + n_{\text{con}}) \quad (5)$$

在本文的目标网络场景下，其主机数为 8，子网

数为 5，不同子网中最大主机数为 5，网络配置信息类型数为 13，因此其状态维度 x=207。此外，对于 $1 \leqslant k \leqslant x$，欺骗防御方状态向量的第 k 维分量 $d^{(k)} \in \{0,1\}$。当 $d^{(k)}=1$ 时，s_i' 的第 k 维对应的网络配置信息为 True；当 $d^{(k)}=0$ 时，s_i' 的第 k 维对应的网络配置信息为 Flase。例如，s_i' 的第 $11N$ 维、第 $12N$ 维以及第 $13N$ 维信息分别对应第 N 台主机此刻是否被妥协、是否可到达以及是否被发现，若第 1 台主机此刻未被攻击者妥协，但可到达已被攻击者发现，此时 s_i' 中的第 11 维~第 13 维的状态向量为 $(0,1,1)$。

定义 2　$A_d=\{a_1',a_2',\cdots,a_i'\}$ 为欺骗防御方的动作集合，即欺骗动作，其中 a_i' 对应于攻击者在状态 s_i 下被防御方欺骗所采取的动作，由一个 z 维的整型集合 c 表示。维度 z 由目标网络的主机数 n_{host}、扫描动作数 n_{scan}、漏洞运行服务类型数 n_{service} 和权限提升进程数 n_{process} 确定。其中，扫描动作包括对每台主机的服务扫描、操作系统扫描、子网扫描、进程扫描这 4 个固定动作，因此 $n_{\text{scan}}=4$。

$$z=n_{\text{host}}(n_{\text{scan}}+n_{\text{service}}+n_{\text{process}}) \tag{6}$$

集合 c 中包含了攻击者可能执行的所有动作，例如，本文的目标网络场景中共有 8 台主机，所有主机包括 2 种主机操作系统：Windows 和 Linux，3 种漏洞运行服务：SSH、HTTP 和 FTP，2 种权限提升进程过程：Tomcat 和 Daclsvc，以及 2 种防火墙限制服务：HTTP 和 SSH。因此，根据定义 2 可知，攻击者采取的动作可以有 72 种操作组合，因此 $c=\{0,1,2,\cdots,71\}$。

定义 3　$R_d=\{r_1',r_2',\cdots,r_i'\}$ 为欺骗防御方的奖励集合，即欺骗奖励，其中 r_i' 对应于攻击者在状态 s_i 下执行动作 a_i 所获得的欺骗奖励，由一个 4 维的整型集合 p 表示，例如 $p=\{-100,0,100,200\}$，不同的取值代表对攻击者不同的欺骗奖励。

3　面向基于强化学习的智能渗透攻击的欺骗防御方法

本节将从攻击者的状态、动作以及奖励数据为切入点，分别从多阶段渗透攻击的前期、中期及后期介绍面向智能渗透测试的欺骗防御方法，欺骗防御模型如图 1 所示。

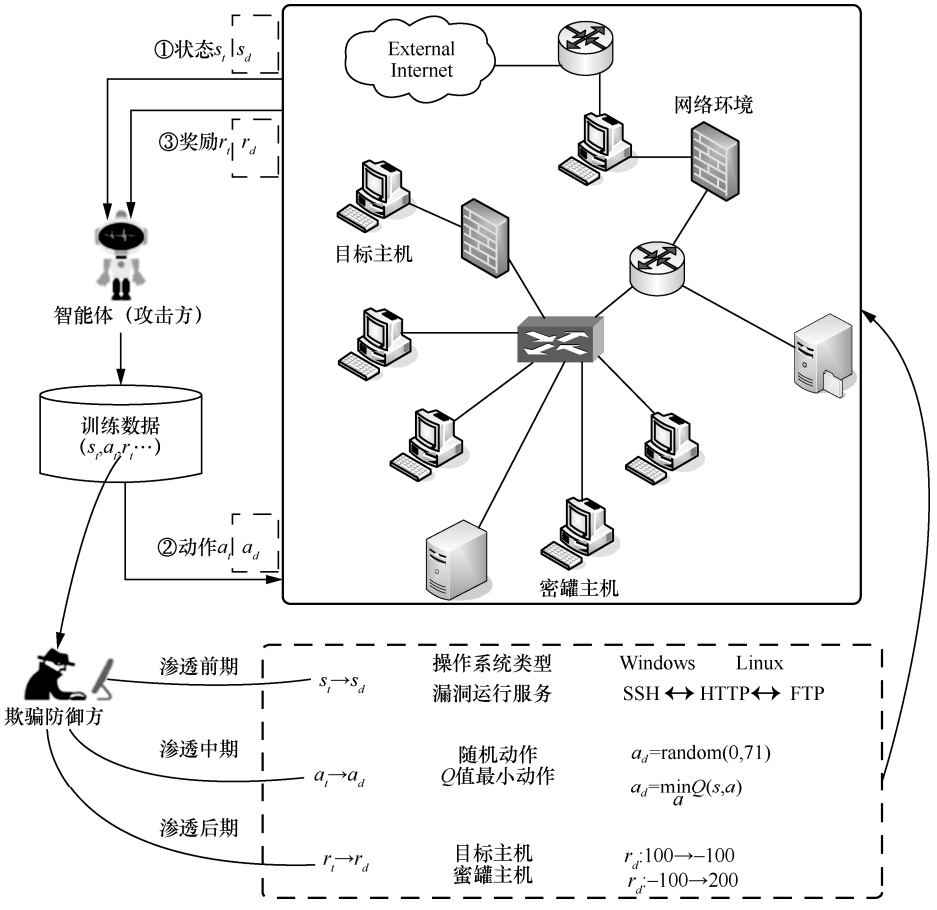


图 1　欺骗防御模型

欺骗防御方通过捕获攻击者训练池中的状态、动作及奖励数据并对其分别修改后，将欺骗数据与攻击者正常训练数据替换，如将攻击者的状态替换成欺骗状态，以此达到欺骗防御的目的。

3.1 状态欺骗防御

渗透攻击前期是多阶段渗透攻击的发起阶段，也是其得以进行的前提和基础。因此，有效防御渗透攻击前期的攻击行为对于抵抗多阶段渗透攻击起着至关重要的作用。在利用 MDP 对基于 QL 的渗透攻击过程建模时，攻击者从目标网络环境中获取到的状态包括目标网络中每台主机的地址、该主机详细配置信息，如该主机是否可到达、是否被发现以及是否被渗透等，此外，还包括该主机的实际渗透价值以及其漏洞服务信息和提权进程所组成的多维布尔向量。攻击者状态获取的过程相当于其在渗透攻击前期所采取的操作。

因此，从防御方角度来看，如何保护目标网络信息不被攻击者完全获取成为抵抗渗透攻击前期攻击行为的关键。本文设计在渗透攻击前期的扫描阶段攻击者获取主机操作系统类型和漏洞运行服务时，通过掩盖主机的真实配置信息，使攻击者扫描不到主机的真实配置信息或扫描出错，如将主机操作系统 Linux 扫描成 Windows；而对于不同的漏洞服务，如 SSH、HTTP 及 FTP 出现扫描不全或者扫描出错的情况。具体而言，对状态进行欺骗防御包括 3 种情况：1) 掩盖操作系统类型；2) 掩盖漏洞运行服务；3) 同时掩盖操作系统类型和漏洞运行服务。

在对攻击者的状态欺骗防御过程中，涉及对操作系统类型和漏洞运行服务的掩盖，因此对于攻击者状态 s_i，只需对其中的第 Nk 维分量 $d^{(Nk)}$ 进行取反操作 RET 即可得到欺骗状态 s_i'。

对应于情况 1)，$k \in [16,17]$，若 $d^{(16N,17N)} = \{1,0\}$，表明此时攻击者扫描到第 N 台主机的操作系统为 Linux；若 $d^{(16N,17N)} = \{0,1\}$，则表明第 N 台主机的操作系统为 Windows，对状态分量 $d^{(16N,17N)}$ 进行按维取反操作，将使攻击者在扫描目标主机真实操作系统时出错，从而达到掩盖真实操作系统类型的目的。

对应于情况 2)，$k \in [18,20]$，d^{18N}、d^{19N} 及 d^{20N} 三维分量分别对应第 N 台主机是否运行的 3 种漏洞服务 SSH、HTTP 及 FTP。若 $d^{(18N,19N,20N)} = \{1,0,1\}$，表明第 N 台主机正运行 SSH 和 FTP 这 2 种漏洞服务，同理，对状态分量 $d^{(18N,19N,20N)}$ 进行按维取反操作，将使攻击者扫描目标主机运行的漏洞服务出错；若 $\mathrm{RET}d^{(18N,19N,20N)} = \{0,1,0\}$，表明第 N 台主机正在运行的 SSH 和 FTP 漏洞服务均未被扫描到，而扫描到该主机正在运行 HTTP 漏洞服务，以此达到掩盖真实漏洞运行服务的目的。

对应于情况 3)，$k \in [17,22]$，为了达到同时掩盖操作系统类型和漏洞运行服务的目的，只需同时对第 N 台主机的五维分量 $d^{(Nk)}$ 进行按维置反操作。综上，状态欺骗防御算法的伪代码如算法 1 所示。

算法 1 状态欺骗防御算法

输入 攻击者状态 S，训练回合数 $i = 20\,000$，主机数 n_{host}，状态维度 k

初始化 $S_d = S$，$i = 1$

1)循环
2) for i =0,1,2,…, do
3)判断
4) if 掩盖主机操作系统类型：
5) $k = \{16,17\}$
6) elif 掩盖主机漏洞服务类型：
7) $k = \{18,19,20\}$
8) else 同时掩盖主机操作系统和漏洞服务：
9) $k = \{16,17,18,19,20\}$
10) end if
11) 结束判断
12)取出对应维度的状态分量 $d^{(Nk)} = S[k]$
13) 按维取反 $d^{(Nk)'} = \mathrm{RET}d^{(Nk)}$
14) 生成欺骗状态 $S_d[k] = d^{(Nk)'}$
15) 以欺骗状态更新攻击者状态 $S \leftarrow S_d$
16)更新当前回合 $i \leftarrow i+1$
17) 返回当前欺骗状态 S_d
18) end for
19)结束循环

3.2 动作欺骗防御

渗透攻击中期是攻击者实施行动的正式阶段，它将直接影响攻击者的渗透结果。因此，如何准确检测攻击者所处位置并干扰其攻击进程将是防御渗透攻击中期攻击行为的关键。

在基于强化学习的智能渗透攻击过程中，本文主要采用干扰攻击者攻击进程的方式来达到欺骗防御的目的。由定义 2 可知，在本文的目标网络下，攻击者采取的动作存在 72 种组合。例如，第 16 个动作指攻击者对子网 2 中的敏感主机执行 Tomcat

进程的权限提升操作，该操作是渗透该网络场景中十分关键的动作，因为子网 2 内的敏感主机是该网络场景中第一台存在可利用漏洞服务的主机，攻击者一旦将其攻破，便可通过权限提升操作获得对该主机的管理权限，进而通过横向移动不断逼近目标主机。

因此，要实现在渗透中期对攻击者的防御，关键在于干扰其采取具体渗透操作的过程。本文通过以下 2 种干扰攻击者动作生成方式进行欺骗防御。

1) 干扰攻击者选择随机动作

$$a_d = \text{random}(0, z-1) \tag{7}$$

其中，z 表示攻击者可采取的动作数，已于定义 2 中给出。

2) 干扰攻击者选择 Q 值最小的动作

$$a_d = \arg\min_a Q(s_t, a) \tag{8}$$

攻击者在采取动作的过程中，令其采取随机动作是最直接的干扰方式，如式(7)所示。具体而言，此时攻击者不会根据在渗透前期扫描到的漏洞信息采取对应的攻击操作，而是以随机动作的方式生成攻击策略，因此该攻击策略存在一定的随机性，进而导致攻击者可能无法成功渗透至最终目标。此外，本文采取另一种干扰程度更高的方法，如式(8)所示，a_d 表示攻击者在 t 时刻状态 s_t 时以 Q 值最小的训练方式所生成的干扰动作。在攻击者正常的攻击策略生成过程中，QL 算法为每一时刻的状态和动作 (s,a) 建立 Q 表，并且每一组状态动作对都对应不同的 Q 值，在训练过程中选取 Q 值最大的动作为训练最优解。为了达到干扰攻击者策略生成的目的，本文通过将 Q 值最大的训练方式修改为 Q 值最小，使攻击者生成的策略无法达到攻击效果，最终实现欺骗防御。动作欺骗防御算法的伪代码如算法 2 所示。

算法 2　动作欺骗防御算法

输入　训练回合数 $i = 20\,000$，主机数 n_{host}，操作系统类型数 n_{os}，漏洞运行服务类型数 n_{service}，不同子网间防火墙限制的服务类型数 n_{firewall}

初始化　Q 表 $Q(s,a)$，学习率 $\alpha \in (0,1]$，折扣因子 $\gamma \in (0,1)$，$i = 1$

1)循环
2)　for i =0,1,2,…, do
3)　初始化攻击者状态 s
4)　基于 e-greedy 贪婪策略从 Q 表派生的 Q 值最大策略中选择 s 对应的动作 a
5)　执行动作 a，观察攻击者奖励 r 及下一个状态 s'
6)　判断
7)　if 采取随机动作
8)　以式(7)生成欺骗动作
9)　else 采取 Q 值最小动作
10)　以式(8)生成欺骗动作
11) end if
12) 结束判断
13)　以欺骗动作替换攻击者动作 $a = a_d$
14)　执行动作 a
15)　观察攻击者奖励 r 及下一个状态 s'
16)　以式(2)更新 Q 表
17)　更新攻击者状态 $s \leftarrow s'$
18)　更新当前回合数 $i \leftarrow i+1$
19)　返回攻击者下一个状态 s'
20)　end for
21)　结束循环

3.3　奖励欺骗防御

在渗透攻击后期，攻击者已经锚定了目标主机并可能采取了相应的攻击操作。针对攻击者的行为，防御方的重点是实现对攻击目标的特殊防护。鉴于奖励值的设定是强化学习算法训练的关键环节，可用于强化攻击者学习到的渗透策略，因此，本文在训练阶段以修改奖励值的方式对其进行欺骗防御，以奖励值翻转的方式强化错误的渗透策略，使其难以寻找到最优的渗透攻击路径，从而无法到达最终目标。

在该阶段，考虑到蜜罐的存在是在渗透中期引诱攻击者的一种方式，因此该阶段不考虑对蜜罐的部署，而是将其作为一类特殊目标来辅助防御过程。因此，本节对是否存在蜜罐网络场景的奖励欺骗防御方法进行讨论。

在无蜜罐网络场景中，本文将敏感主机的奖励值符号进行翻转来降低攻击者的渗透成功率，即将敏感主机的价值由 100 修改为−100，同时将训练过程每个回合的结束条件修改为回合训练步数达到设置的最大值 1 000 步，而获得目标敏感主机的 Root 权限时也并不表示回合结束。此时，攻击者在训练过程中即使渗透到了敏感主机，也不会产生渗透成功标志，并且获得的奖励回报很低，使攻击者误以为这种动作策略是无效的渗透策略，从而达到降低攻击者性能的目的，影响敏感主机渗透攻击的成功率。

在有蜜罐网络场景中，本文同样将训练数据的

奖励值符号进行翻转来对攻击者进行欺骗防御，包括蜜罐主机和敏感主机，其中蜜罐主机的奖励值由−100 修改为 200，敏感主机的奖励值修改为−100，这是由于网络中存在 2 个敏感主机，找到所有蜜罐主机获得的奖励值接近于 200。当蜜罐主机的奖励值修改为 200 时，攻击者在欺骗防御训练过程中获得的奖励值训练效果会接近于正常训练，并且陷入蜜罐的概率也将加大。同时将每个回合的结束条件修改为：获得蜜罐主机的 User 权限，且回合训练步数达到了设置的最大值。奖励欺骗防御算法的伪代码如算法 3 所示。

算法 3 奖励欺骗防御算法

输入 攻击者当前状态 s，训练回合数 $i=20\,000$

初始化 Q 表 $Q(s,a)$，欺骗奖励池 $p=\{-100,0,200\}$，学习率 $\alpha\in(0,1]$，折扣因子 $\gamma\in(0,1)$，$i=1$

1)循环
2) for i =0,1,2,…, do
3) 初始化攻击者状态 s
4) 基于 e-greedy 贪婪策略从 Q 表派生的 Q 值最大策略中选择 s 对应的动作 a
5) 执行动作 a，观察攻击者奖励 r 及下一个状态 s'；
6) 判断
7) if $r=100$
8) 欺骗奖励 $r_d=p[0]$
9) elif $r=-100$
10) 欺骗奖励 $r_d=p[2]$
11) else
12) 欺骗奖励：$r_d=p[1]$
13) end if
14) 结束判断
15) $r=r_d$
16) 以式(2)更新 Q 表
17) 更新攻击者状态：$s\leftarrow s'$
18) 更新当前回合数：$i\leftarrow i+1$
19) 返回攻击者下一个状态 s'
20) end for
21)结束循环

4 实验结果与分析

4.1 实验环境设置

实验环境的具体配置如下：CPU 型号为 iE3-123v3@3.40 GHz，运行内存为 32 GB，操作系统为 Ubuntu 16.04，编程语言为 Python 3.7.10，深度学习框架为 PyTorch-1.5.0。本文在 NASim 平台[28]构建的模拟网络环境中进行智能渗透攻击和欺骗防御研究，利用 QL 算法[22]进行实验，共训练 20 000 个回合，并在每个回合都记录渗透攻击路径生成所需要的时间步数和成本消耗，在训练阶段中每隔 1 000 个回合计算一次平均值，采用的评价指标如下。

1) 平均回合奖励。在训练过程中，每隔 1 000 个回合计算一次攻击者的平均奖励值，用于评估攻击者的渗透性能。

2) 平均回合步数。在训练过程中，每隔 1 000 个回合计算一次平均回合长度，表示攻击者渗透攻击路径生成所需的时间成本，用于评估攻击者的渗透性能。

3) 陷入蜜罐概率。在有蜜罐网络场景进行渗透攻击和欺骗防御过程中，每隔 1 000 个回合计算一次蜜罐主机入侵的平均概率，用于评价攻击和防御效果。

4.2 目标网络场景介绍

本文针对图 2 所示的目标网络场景进行渗透攻击，基于 QL 算法训练攻击者寻找最优渗透路径并实现智能渗透攻击。

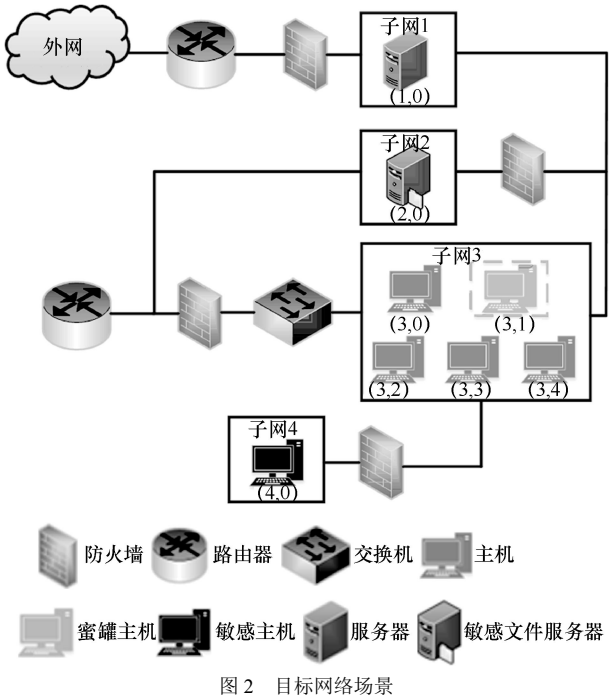


图 2 目标网络场景

目标网络场景中包含 4 个子网络、8 台主机和服务器，其中有 2 台为敏感主机，分别为(2,0)和(4,0)，其中(4,0)为目标主机，该 2 台主机均运行了不同的漏洞服务。只有子网 1 可以和外网直接通信，不同子网间的

通信被防火墙所限制的不同服务所阻隔，每次从一个子网渗透到另一个子网都需要消耗一定的成本。此外，子网内的主机可以相互通信，子网 1 可以直接和子网 2 及子网 3 进行通信，但不能与子网 4 进行通信，所以攻击者想要成功渗透目标敏感主机(4,0)，就必须先渗透攻击成功子网 2 或子网 3 内的主机，通过提权操作获得渗透成功主机的 Root 权限后，再利用横向移动对子网 4 的目标敏感主机进行渗透攻击，直至渗透结束。

主机配置信息如表 1 所示，包括主机地址、操作系统、漏洞服务、权限提升进程以及主机价值。为了模拟真实攻击者的行为，本文假设攻击者无法直接获取网络的拓扑信息和主机配置的相关信息，因此，除了漏洞利用（Exp, exploit）和权限提升（PE, privilege escalation）动作之外，还可以采用扫描动作获取目标网络主机的相关信息。对于每台主机来说，攻击者可以选取表 2 中的动作，本文选用子网渗透过程中常被攻击者利用的进程和服务来替代网络安全漏洞。例如，攻击者可以通过 FTP 漏洞来得到主机的 User 权限，利用 Daclsvc 权限提升进程来获得主机的 Root 权限，从而实现对目标主机的渗透攻击。

表 1　　主机配置信息

主机地址	操作系统	漏洞服务	权限提升进程	主机价值
(1,0)	Linux	HTTP	Tomcat	0
(2,0)	Linux	SSH，FTP	/	100
(3,0)	Windows	FTP	/	0
(3,1)	Windows	FTP，HTTP	Daclsvc	0
(3,2)	Windows	FTP，HTTP	Daclsvc	0
(3,3)	Windows	FTP	/	0
(3,4)	Windows	FTP	Daclsvc	0
(4,0)	Linux	SSH，FTP	Tomacat	100

表 2　　攻击者动作

名称	类型	操作系统	成本消耗	概率	访问权限
SSH	渗透	Linux	3	0.9	User
FTP	渗透	Windows	1	0.6	User
HTTP	渗透	/	2	0.9	User
Tomcat	提权	Linux	1	1	Root
Daclsvc	提权	Windows	1	1	Root
服务扫描	扫描	/	1	1	/
操作系统扫描	扫描	/	1	1	/
子网扫描	扫描	/	1	1	/
进程扫描	扫描	/	1	1	/

此外，本文还研究了对蜜罐网络的渗透攻击，即将图 2 中(3,1)的普通主机替换为蜜罐主机。在传统的渗透测试中，蜜罐可以伪装成敏感主机来诱使渗透攻击者对其进行渗透，从而有效保护网络中的敏感主机。但是，在正常渗透攻击该场景时，由于攻击者事先知晓蜜罐的地址，并且蜜罐主机所赋价值为−100，所以攻击者经过训练后会以绕过蜜罐主机的方式对其余子网的目标主机进行渗透攻击。

因此，在对攻击者的状态和动作进行欺骗防御的情况下，本文不对蜜罐主机的价值做修改处理，此时的蜜罐设置起不到辅助防御的作用，若攻击者陷入蜜罐主机的概率提升，则恰巧说明是状态或动作的干扰起到了欺骗攻击者的效果，从而导致其陷入其中。为使防御实验更充分，本文在有蜜罐的目标网络场景中进行状态和动作欺骗防御。相反，由式(3)可知，攻击者获得的奖励与主机价值和渗透该主机的成本有关，在以修改主机价值对攻击者进行奖励欺骗防御的过程中，通过对蜜罐主机原本的低价值翻转为高价值即可将该蜜罐主机伪装成目标敏感主机，从而达到欺骗攻击者的目的；同理，将敏感主机的高价值翻转为低价值，攻击者将敏感主机误以为蜜罐主机，并选择绕开该主机，不对其采取渗透操作，以此达到保护敏感主机的目的。因此，本文分别在无蜜罐和有蜜罐的网络中展开奖励欺骗防御实验。

4.3　状态欺骗防御时渗透攻击性能分析

在渗透攻击者扫描阶段，本文通过掩盖主机操作系统类型及漏洞运行服务干扰攻击者对目标网络正常状态的获取，如将主机操作系统 Linux 扫描成 Windows，而对不同的 SSH、FTP 及 HTTP 漏洞服务扫描不全或者扫描出错，以此达到掩盖真实网络信息的目的。此外，针对在渗透前期常采用的地址跳变[17-18]欺骗防御方法，考虑到主机地址是攻击者状态中的第 Nk 维分量 $d^{(Nk)}$，$k\in[0,9]$，本节结合地址跳变的思想，同样采取对应维度置反的方法使攻击者扫描到错误的 IP 地址，并作为上述另外 3 种状态欺骗防御方法的对照基准。本节以 4 组不同的数据干扰比例（20%、50%、80%及 100%）分别对主机地址跳变、主机操作系统、主机漏洞服务以及对主机操作系统和漏洞服务均欺骗掩盖的 4 种方法在有蜜罐的目标网络下进行对比实验，每组对比实验均训练 20 000 回合。

对于 4 种状态欺骗方法而言，从图 5~图 8 可以发现一个共同现象：干扰比例越高，欺骗防御效果越好，当干扰比例超过 50%时，欺骗防御效果愈加明显。对

于主机地址跳变、干扰主机漏洞服务以及同时干扰主机操作系统和漏洞服务的 3 种状态欺骗方法而言，图 5、图 7、图 8 存在一个共同现象：当干扰比例为 20%时，从平均回合奖励及平均回合步数来看其干扰效果并不明显，攻击者均能在 6 000 回合左右寻找到最优路径，训练成本较正常训练只增加 1 000 回合。这也从另一方面说明强化学习模型以不断试错的训练方式在 20%的低干扰比例内，能通过“自愈”的方法恢复到正常性能。

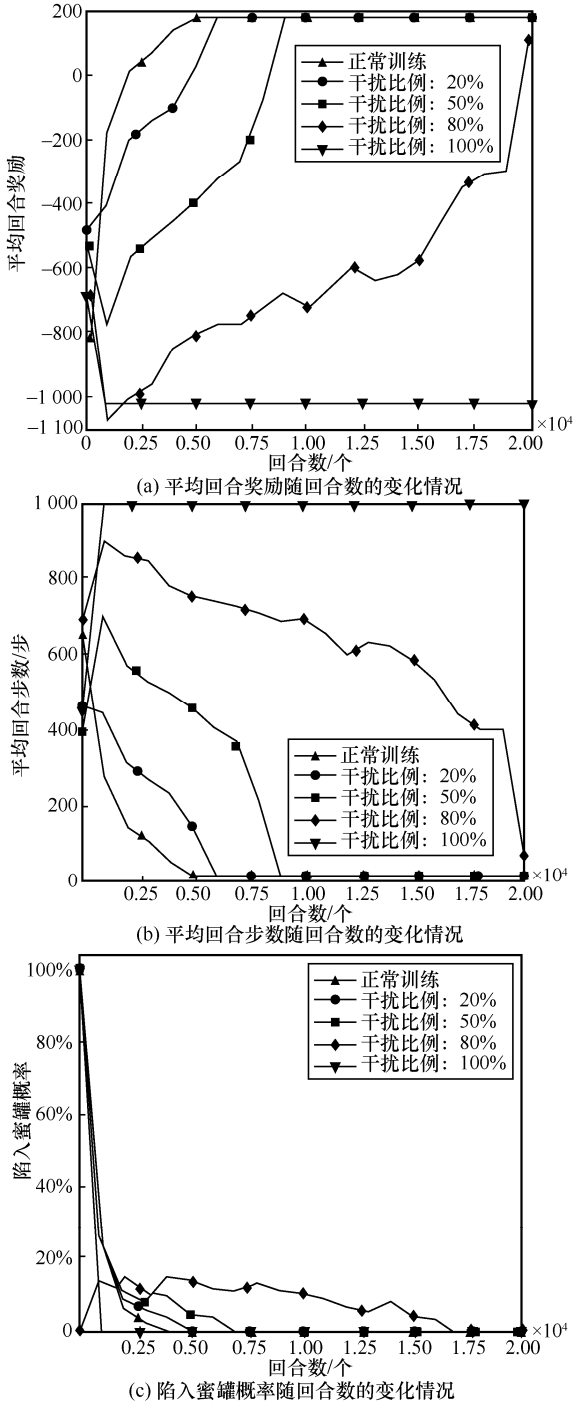


图 5　对主机地址跳变进行欺骗防御训练效果随回合数的变化情况

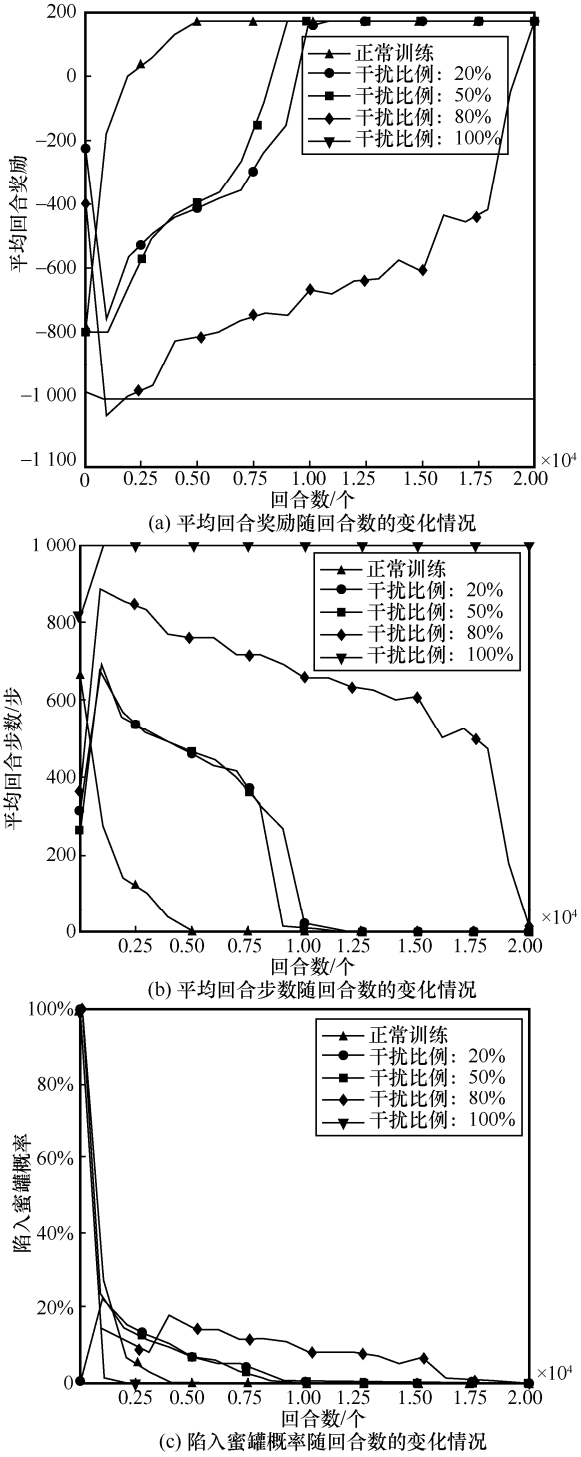
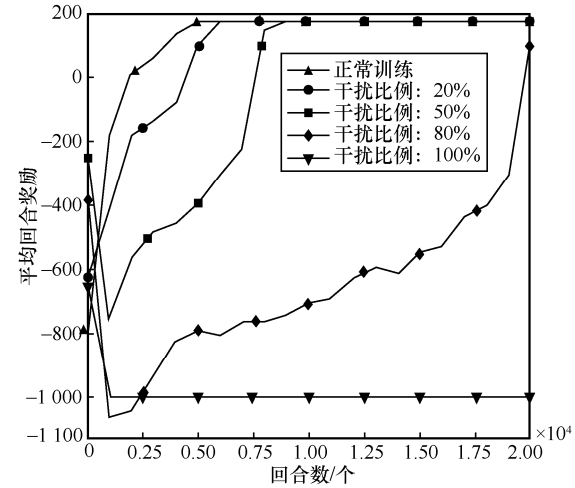


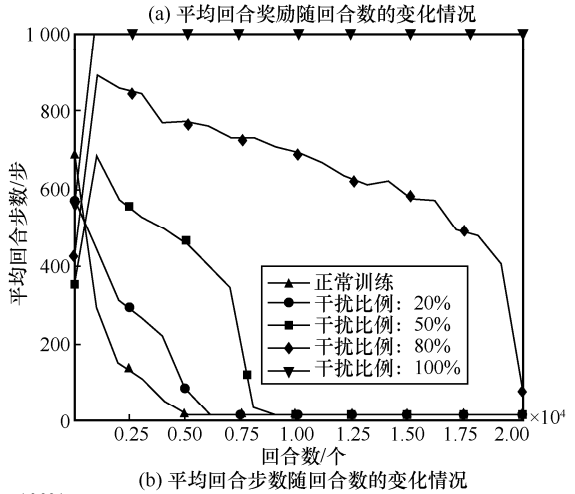
图 6　对主机操作系统进行欺骗防御训练效果

对于主机操作系统而言，当干扰比例为 20%时，其防御性能接近其余 3 种状态欺骗方法在干扰比例为 50%时的性能，训练成本较正常训练增加近 4 000 回合，这也说明通过干扰主机操作系统可以低干扰成本达到较高防御性能。攻击者通过渗透与主机操作系统相匹配的漏洞服务获取其相应的管理权限，只有在成功渗透该主机漏洞服务的前提下

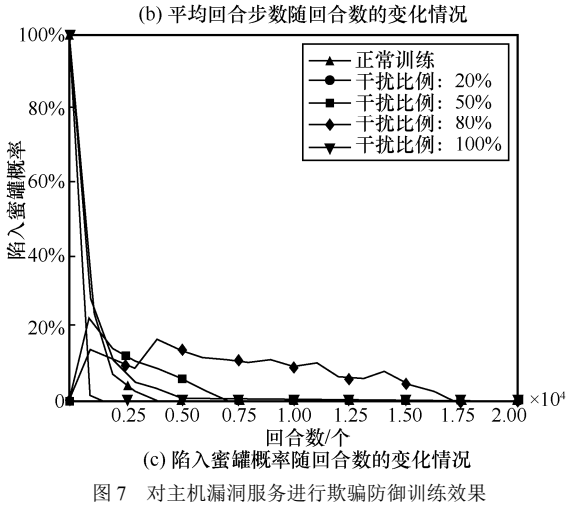
才能进行提权操作；一旦主机操作系统受到干扰，将导致主机操作系统与漏洞服务不匹配，攻击者可能无法对该主机采取与其漏洞服务相匹配的渗透方式，最终导致无法成功渗透目标主机。



(a) 平均回合奖励随回合数的变化情况



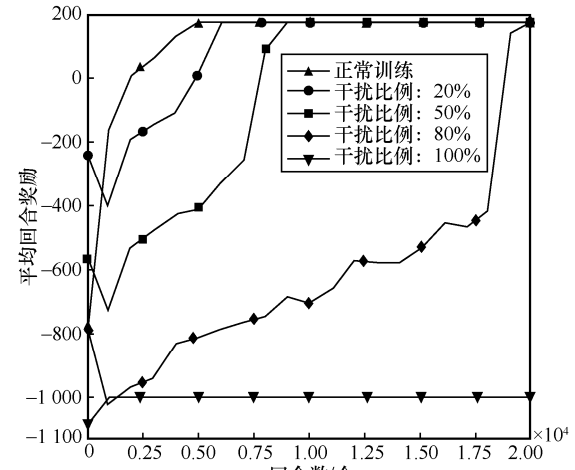
(b) 平均回合步数随回合数的变化情况



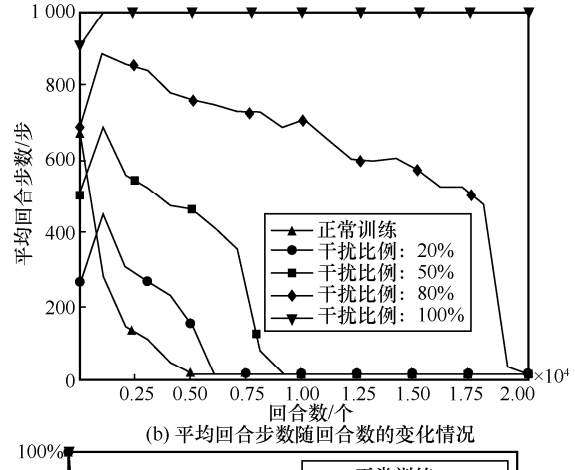
(c) 陷入蜜罐概率随回合数的变化情况

图 7　对主机漏洞服务进行欺骗防御训练效果

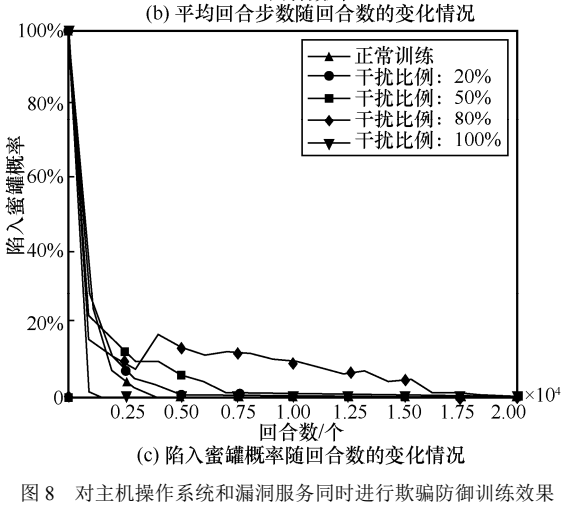
当干扰比例扩大至 50%~80%时，对 4 种状态欺骗方法的防御效果都比较明显，其训练成本较正常训练从 10 000 回合增加至 20 000 回合，随着干扰比例的扩大，训练成本增加了一倍，当干扰比例为 80%时，就已经达到了很好的欺骗防御效果，4 种状态欺骗方法均只能在训练的最终阶段接近正常训练时的性能，而对于地址跳变和主机漏洞服务 2 种干扰方式，攻击者在后期也无法成功渗透至目标主机。而当干扰比例为 100%时，无论从平均回合奖励还是平均回合步数来看，攻击者均不能找到最优的渗透路径，每个回合均以失败告终。



(a) 平均回合奖励随回合数的变化情况



(b) 平均回合步数随回合数的变化情况



(c) 陷入蜜罐概率随回合数的变化情况

图 8　对主机操作系统和漏洞服务同时进行欺骗防御训练效果

对于陷入蜜罐概率而言，图 5(c)、图 6(c)、图 7(c)以及图 8(c)呈现出较一致的变化趋势。当干扰比例为 50%时，4 种状态欺骗方法的陷入蜜罐概率在 8 000 回合左右逐渐降为 0；当干扰比例为 80%时，陷入蜜罐概率在训练后期 17 000 回合才逐渐降为 0；当干扰比例为 100%时，陷入蜜罐概率在 2 500 回合就已经降为 0，这是因为在高干扰比例下，攻击者一开始就无法渗透成功，从而也不能陷入子网 3 中的蜜罐，这也从另一方面验证了蜜罐部署的防御手段不适合在渗透前期进行这一观点。

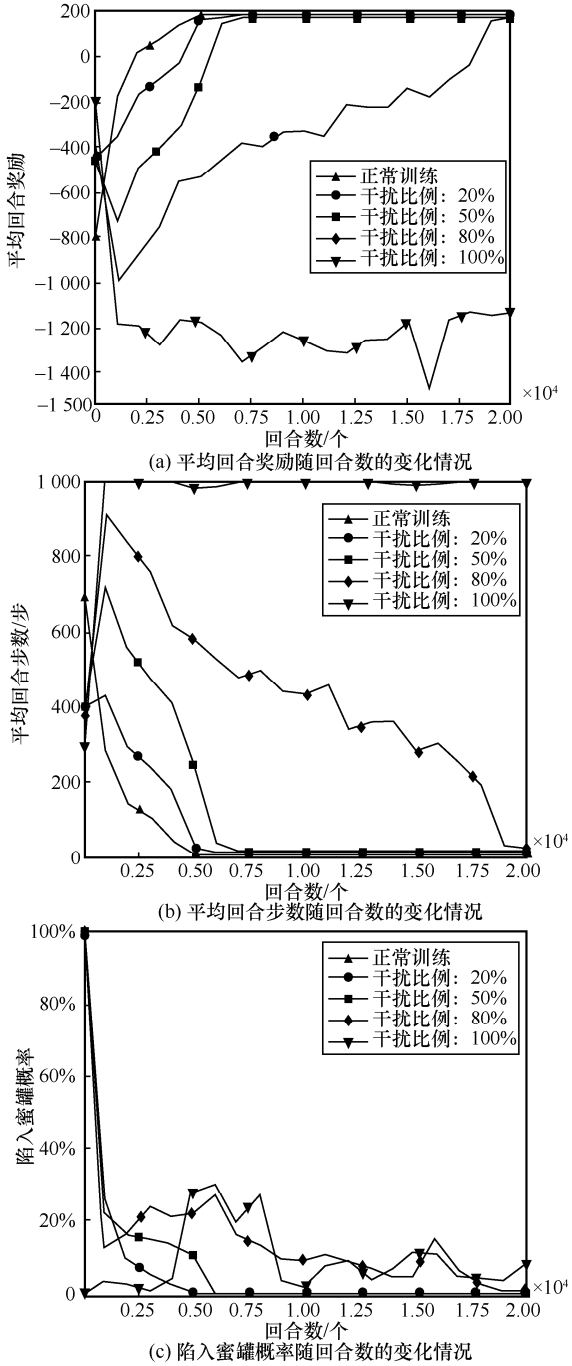
从图 5~图 8 中可以直观地发现，在不同的干扰比例下不同的状态欺骗方法拥有不同的优势。具体而言，当干扰比例在 50%以内时，干扰主机操作系统的欺骗方法防御性能最佳；当干扰比例为 80%时，干扰主机漏洞服务和主机地址跳变这 2 种欺骗方的防御效果最好，在最终回合 2 种欺骗方法均使攻击者无法成功渗透至最终目标主机；当干扰比例为 100%时，4 种方法的防御性能相当。此外，相比于主机地址跳变，干扰主机操作系统和漏洞服务的方法更贴近真实渗透场景，可操作性强；而主机地址跳变方法虽然本身具有一定的防御效果，但其需对状态的前 10 维信息均进行置反操作，且难以对干扰后状态的实际含义进行合理解释。

综合上述 4 种不同的状态欺骗防御方法可知，无论是仅干扰一种网络要素，或是同时对 2 种关键网络要素进行干扰，均较难达到以低干扰成本实现完全防御的目的，状态获取是攻击者执行渗透的前期阶段，在后续的中期及后期阶段同样也存在受到欺骗的可能，并且不同阶段采取不同的欺骗方法也将产生不同的防御效果。

4.4　动作欺骗防御时渗透攻击性能分析

在渗透攻击前期和中期，攻击者获取到目标网络的状态信息后执行实际攻击动作时，通过扰乱攻击者动作生成的方式达到欺骗防御的目的。本节以干扰攻击者采取随机生成动作和干扰攻击者采取 Q 值最小动作的方式在有蜜罐的目标网络中进行欺骗防御，以 4 组不同的动作数据干扰比例（20%、50%、80%、100%）进行对比实验，每组对比实验均训练 20 000 回合，并以随机动作的干扰方式作为对照基准。

如图 9 所示，欺骗攻击者采取随机动作的方式在干扰比例在 50%以内时的防御效果与状态欺骗防御效果相似，攻击者以不断试错的训练方式并通过其“自愈”的方法恢复到正常性能。当干扰比例扩大到 80%时，其平均回合奖励和平均回合步数均发生大幅度变化，攻击者只能在后期找到渗透目标。当干扰比例为 100%时，平均回合奖励在 2 000 回合后就开始呈现高负值且不断波动的状态，而其平均回合步数居高不下，这反映出攻击者始终无法找到最优渗透路径的局面。此外，从陷入蜜罐概率的情况来看，当干扰比例大于 80%时，随机动作的干扰在 3 000 回合左右呈现小幅上升的趋势，此时说明随机动作的干扰对引诱攻击者陷入蜜罐起到一定的正向作用。



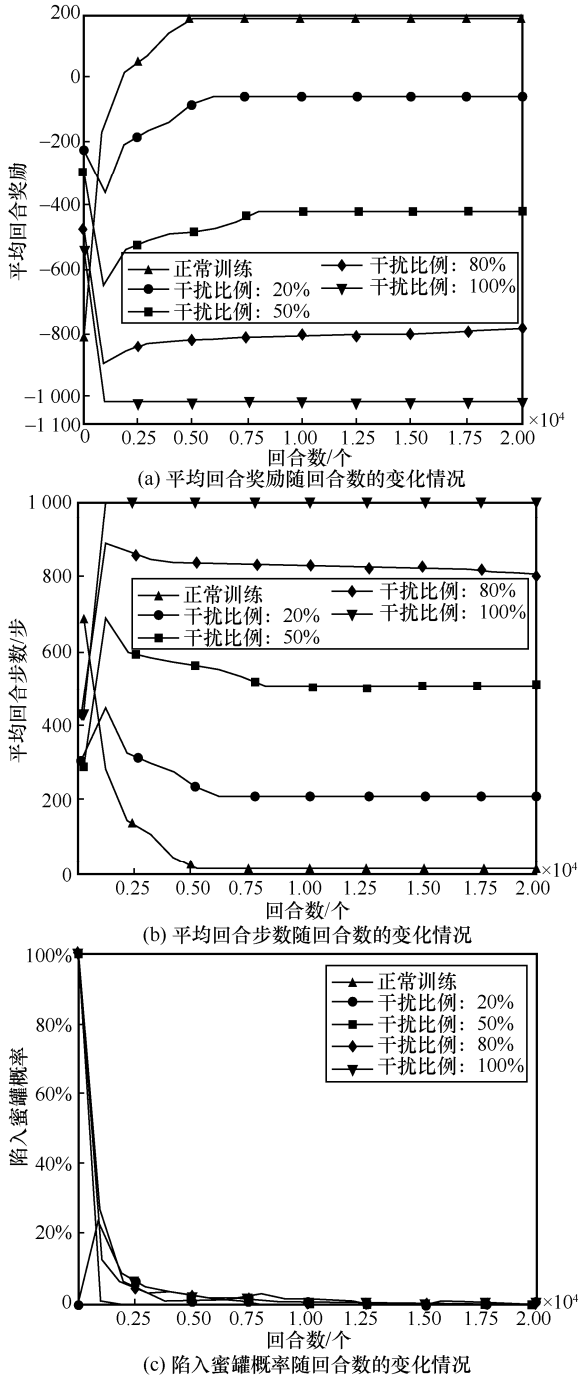
(a) 平均回合奖励随回合数的变化情况

(b) 平均回合步数随回合数的变化情况

(c) 陷入蜜罐概率随回合数的变化情况

图 9　对攻击者采取随机动作进行欺骗防御训练效果

采用干扰攻击者选择 Q 值最小动作的欺骗防御方法时，如图 10(a)和图 10(b)所示，与之前所采用的攻击方法呈现出完全不同的结果。首先，当干扰比例在 20%时，渗透攻击成功率已降低为 0，此时攻击者的平均回合奖励值已为负值，平均回合步数也增加至 200 步。其次，随着干扰比例的增加，平均回合负值奖励及平均回合步数均在不断增加，这说明干扰比例越高，欺骗防御攻击者的效果越好。



(a) 平均回合奖励随回合数的变化情况

(b) 平均回合步数随回合数的变化情况

(c) 陷入蜜罐概率随回合数的变化情况

图 10　对攻击者采取 Q 值最小动作进行欺骗防御训练效果

对于陷入蜜罐概率而言，如图 10(c)所示，4 种不同干扰比例下的陷入蜜罐概率在训练前期就已全部为 0，这并不代表蜜罐的部署没有效果，而是说明攻击者在训练前期就无法执行正常的渗透过程，从而也无法渗透到达蜜罐主机所在的位置。这也说明了干扰攻击者动作选择的方式在训练前期就能达到很好的欺骗防御效果。

总之，在渗透中期通过干扰动作的方式对攻击者进行欺骗可实现以低成本达到高防御性能的效果，相比于随机动作的干扰方式，以干扰攻击者采取 Q 值最小动作的方式防御性能最好，但该方法的弊端在于需事先获得攻击者的训练模型，也属于白盒渗透的一种方式。本文继续在渗透后期以对攻击目标进行特殊防护的角度展开欺骗防御研究。

4.5　奖励欺骗防御时渗透攻击性能分析

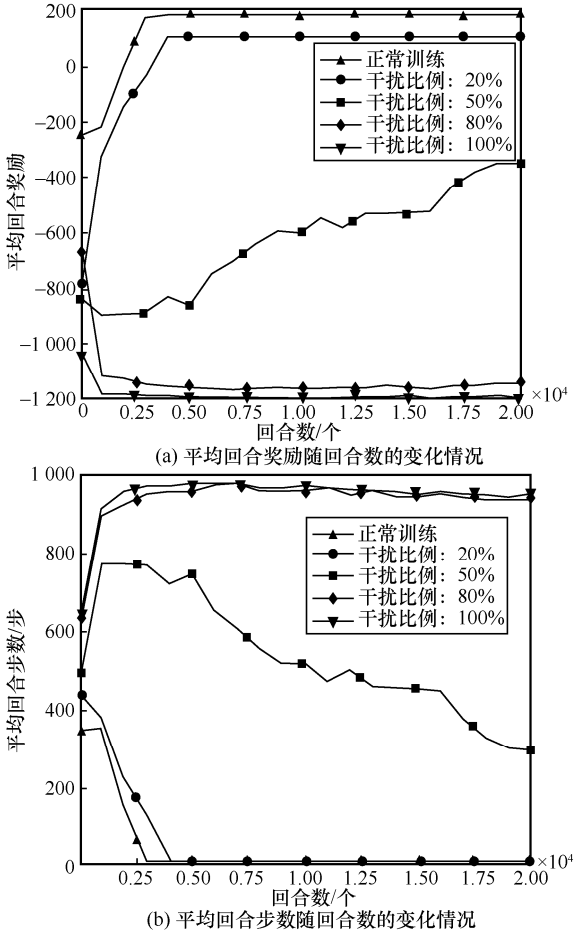
本节以奖励值符号翻转的方式展开奖励欺骗防御研究。具体而言，在无蜜罐网络场景下，将敏感主机的价值由 100 修改为−100；在有蜜罐网络场景下，为了达到训练攻击者不断陷入蜜罐的同时奖励值不影响模型整体性能的效果，将蜜罐主机的价值修改为 200，而敏感主机的价值修改为−200。本节分别在无蜜罐网络和有蜜罐网络 2 种场景进下行对比实验，并同样以 4 组不同的奖励数据干扰比例（20%、50%、80%及 100%）进行 20 000 回合训练。

4.5.1　无蜜罐网络场景奖励欺骗防御性能分析

针对攻击者对无蜜罐网络的渗透过程，本节主要通过修改敏感主机的奖励值达到保护敏感主机的目的。无蜜罐场景对奖励数据进行欺骗防御训练效果如图 11 所示。从图 11 中可以发现，随着回合数的增加，数据干扰比例越高，攻击者的平均回合奖励越小，平均回合步数越高，当干扰比例为 80%和 100%时，平均回合步数接近最大值 1 000。由此可知，干扰比例越大，防御性能越好。

此外，本节还记录了测试阶段的渗透攻击效果，表 3 展示了无蜜罐网络场景不同干扰比例下的欺骗防御效果。从表 3 中可以发现，当数据干扰比例为 20%和 50%时，攻击者最终仍可成功渗透敏感主机。当数据干扰比例为 80%和 100%时，攻击者即使在达到最大回合步数的情况下，还是无法对敏感主机进行有效渗透。可见，在无蜜罐网络场景下，通过干扰奖励数据的方式进行欺骗

防御，可以有效扰乱攻击者策略，增加其时间成本和漏洞利用成本使其策略失效，从而达到欺骗防御目的。



(a) 平均回合奖励随回合数的变化情况

(b) 平均回合步数随回合数的变化情况

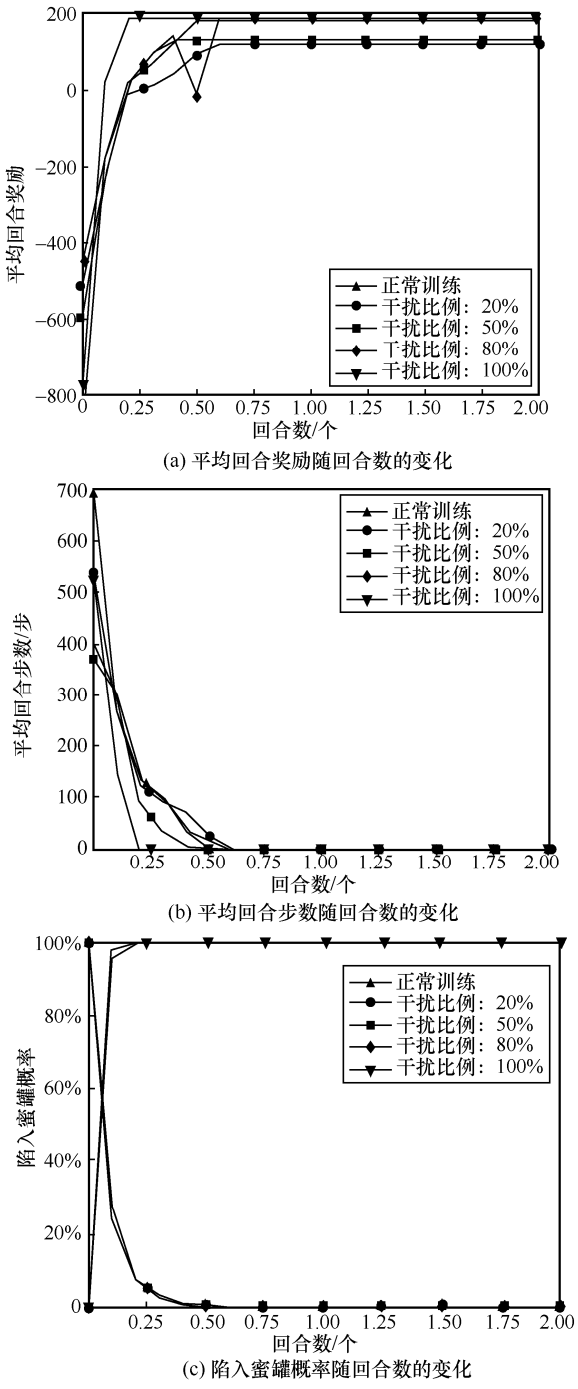
图 11　无蜜罐场景对奖励数据进行欺骗防御训练效果

表 3　无蜜罐网络场景不同干扰比例下的欺骗防御效果

干扰比例	奖励	回合步数	渗透敏感主机概率
20%	180	12	100%
50%	176	15	100%
80%	−1 030	1 000	0
100%	−1 172	1 000	0

4.5.2　有蜜罐网络场景奖励欺骗防御性能分析

本节一方面对敏感主机和蜜罐主机的奖励值进行修改，将蜜罐主机的奖励值设为 200 伪装成敏感主机；另一方面将结束条件修改为获得蜜罐主机的 User 权限或回合步数达到最大值。图 12 展示了有蜜罐网络对奖励数据进行欺骗防御训练效果。比较图 11 与图 12 的蜜罐网络训练效果可发现，在奖励欺骗过程中，攻击者渗透效果与正常渗透效果相差较小。此外，数据干扰比例越大，攻击者最终的奖励值越大。这是由于干扰比例越大，攻击者陷入蜜罐概率越大，且步数越来越短，消耗的成本代价减小，使最终的奖励值变大，欺骗防御效果越好。但该方法的不足之处在于需提前获得敏感主机和蜜罐的地址，而不是利用主动探测或其他手段将这两类主机进行区分。



(a) 平均回合奖励随回合数的变化

(b) 平均回合步数随回合数的变化

(c) 陷入蜜罐概率随回合数的变化

图 12　有蜜罐网络对奖励数据进行欺骗防御训练效果

从图 12(c)可以发现，除了数据干扰比例为 20%的攻击者最终能躲过蜜罐主机的迷惑，其他干扰比例下攻击者都把蜜罐主机作为网络中的脆弱节点进行渗透，进而陷入其中。

表 4 展示了有蜜罐网络场景不同干扰比例下的欺骗防御效果。从表 4 中可以发现，在测试过程中，当干扰比例为 50%、80%和 100%时，攻击者由于在欺骗防御后能很快地找到蜜罐主机，从而回合步数基本很短，并且最终经过欺骗防御后的策略 100%的偏向于陷入蜜罐主机，与正常渗透的趋势吻合。与表 3 相比，有蜜罐网络中干扰比例为 50%时就能达到欺骗防御攻击的目的，比无蜜罐网络中的防御效果更好。

表 4　有蜜罐网络场景不同干扰比例下的欺骗防御效果

干扰比例	奖励	回合步数	渗透敏感主机概率
20%	180	12	2%
50%	−109	5	100%
80%	−112	6	100%
100%	−109	5	100%

5　结束语

本文对基于强化学习的智能渗透攻击进行欺骗防御研究，分别从攻击者的状态、动作及奖励 3 个角度出发，对应于渗透攻击的前期、中期及后期 3 个阶段展开实验分析。针对无蜜罐和有蜜罐的不同目标网络场景，本文首先基于强化学习实现智能渗透攻击。其次，通过修改攻击者状态、动作及奖励训练数据的方式来对攻击者进行欺骗防御。最后，通过实验证明在智能渗透攻击过程中利用污染强化学习模型训练数据的思路，可对攻击者达到很好的欺骗防御效果，其中，干扰攻击者动作生成和修改不同主机奖励值的防御方法的防御性能最显著。

参考文献：

[1] ARKIN B, STENDER S, MCGRAW G. Software penetration testing[J]. IEEE Security & Privacy, 2005, 3(1): 84-87.

[2] 杨宏宇, 袁海航, 张良. 基于攻击图的主机安全评估方法[J]. 通信学报, 2022, 43(2): 89-99.
ROWE N C, CUSTY EJ, DUONG B T. Defending cyberspace with fake honeypots[J]. Journal of Computers, 2007, 2(2): 25-36.

[3] KAUR G, KAUR N. Penetration testing-reconnaissance with Nmap tool[J]. International Journal of Advanced Research in Computer Science, 2017, 8(3): 844-846.

[4] MULIŃSKI T. ICT security in tax administration - Rapid7 Nexpose vulnerability analysis[J]. Studia Informatica, 2021, 24: 37-51.

[5] LEE A. Advanced penetration testing for highly-secured environments: the ultimate security guide[M]. Birmingham: Packt Publishing, 2012.

[6] HelpSysthems. Core impact[EB]. 2021.

[7] SAYED A. Adaptation, learning, and optimization over networks[J]. Foundations and Trends in Machine Learning, 2014, 7(4/5): 311-801.

[8] MNIH V, KAVUKCUOGLU K, SILVER D, et al. Playing atari with deep reinforcement learning[J]. arXiv Preprint, arXiv: 1312.5602, 2013.

[9] ZHOU S C, LIU J J, HOU D D, et al. Autonomous penetration testing based on improved deep Q-network[J]. Applied Sciences, 2021, 11(19): 8823.

[10] TRAN K, AKELLA A, STANDEN M, et al. Deep hierarchical reinforcement agents for automated penetration testing[J]. arXiv Preprint, arXiv: 2109.06449, 2021.

[11] DULAC-ARNOLD G, EVANS R, SUNEHAGP, et al. Reinforcement learning in large discrete action spaces[J]. arXiv Preprint, arXiv: 1512.07679, 2015.

[12] YUILL J J. Defensive computer-security deception operations: processes, principles and techniques[D]. Raleigh: North Carolina State University, 2006.

[13] Gartner Research. Hype cycle for threat-facing technologies 2017[R]. 2017.

[14] 贾召鹏, 方滨兴, 刘潮歌, 等. 网络欺骗技术综述[J]. 通信学报, 2017, 38(12): 128-143.
JIA Z P, FANG B X, LIU C G, et al. Survey on cyber deception[J]. Journal on Communications, 2017, 38(12): 128-143.

[15] 胡永进, 马骏, 郭渊博. 基于博弈论的网络欺骗研究[J]. 通信学报, 2018, 39(S2): 9-18.
HU Y J, MA J, GUO Y B. Research on cyber deception based on game theory[J]. Journal on Communications, 2018, 39(S2): 9-18.

[16] 王硕, 王建华, 裴庆祺,等. 基于动态伪装网络的主动欺骗防御方法[J]. 通信学报, 2020, 41(2): 97-111.
WANG S, WANG J H, PEI Q Q, et al. Active deception defense method based on dynamic camouflage network[J]. Journal on Communications, 2020, 41(2): 97-111.

[17] JAFARIAN J H, AL-SHAER E, DUAN Q. Adversary-aware IP address randomization for proactive agility against sophisticated attackers[C]//Proceedings of 2015 IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2015: 738-746.

[18] WANG K, CHEN X, ZHU Y F. Random domain name and address mutation (RDAM) for thwarting reconnaissance attacks[J]. PLoS One, 2017, 12(5): e0177111.

[19] ANAGNOSTAKIS K, SIDIROGLOU S, AKRITIDIS P, et al. Detecting targeted attacks using shadow honeypots[C]//Proceedings of the 14th Conference on USENIX Security Symposium. Berkeley: USE-

NIX Association, 2005: 9.

[20] ROWE N C, CUSTY E J, DUONG B T. Defending cyberspace with fake honeypots[J]. Journal of Computers, 2007, 2(2): 25-36.

[21] 石乐义, 姜蓝蓝, 刘昕, 等. 拟态式蜜罐诱骗特性的博弈理论分析[J]. 电子与信息学报, 2013, 35(5): 1063-1068.
SHI L Y, JIANG L L, LIU X, et al. Game theoretic analysis for the feature of mimicry honeypot[J]. Journal of Electronics & Information Technology, 2013, 35(5): 1063-1068.

[22] SILVER D, HUANG A, MADDISON C J, et al. Mastering the game of Go with deep neural networks and tree search[J]. Nature, 2016, 529(7587): 484-489.

[23] BERNER C, BROCKMAN G, CHAN B, et al. Dota 2 with large scale deep reinforcement learning[J]. arXiv Preprint, arXiv: 1912.06680, 2019.

[24] VINYALS O, BABUSCHKIN I, CZARNECKI W M, et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning[J]. Nature, 2019, 575(7782): 350-354.

[25] SCHWARTZ J, KURNIAWATI H. Autonomous penetration testing using reinforcement learning[J]. arXiv Preprint, arXiv: 1905.05965, 2019.

[26] ZENNARO F M, ERDODI L. Modeling penetration testing with reinforcement learning using capture-the-flag challenges and tabular Q-learning[J]. arXiv Preprint, arXiv: 2005.12632, 2005.

[27] 臧艺超, 周天阳, 朱俊虎, 等. 领域独立智能规划技术及其面向自动化渗透测试的攻击路径发现研究进展[J]. 电子与信息学报, 2020, 42(9): 2095-2107.
ZANG Y C, ZHOU T Y, ZHU J H, et al. Domain-independent intelligent planning technology and its application to automated penetration testing oriented attack path discovery[J]. Journal of Electronics & Information Technology, 2020, 42(9): 2095-2107.

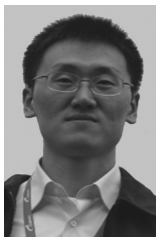
[28] SCHWARTZ, J. Network attack simulator[EB]. 2017.

[作者简介]

陈晋音（1982− ），女，浙江象山人，博士，浙江工业大学教授、博士生导师，主要研究方向为人工智能、数据挖掘、智能计算。

胡书隆（1998− ），男，江西吉安人，浙江工业大学硕士生，主要研究方向为深度强化学习和网络安全。

邢长友（1982− ），男，江苏南京人，博士，陆军工程大学副教授、硕士生导师，主要研究方向为网络安全、软件定义网络、网络测量和网络功能虚拟化。



张国敏（1979− ），男，江苏南京人，博士，陆军工程大学副教授、硕士生导师，主要研究方向为软件定义网络、网络安全、网络测量和网络功能虚拟化。