

基于强化学习的自动化 Windows 域渗透方法

占力戈^{1,2}, 沙乐天^{1,2}, 肖甫^{1,2}, 董建阔^{1,2}, 张品昌^{1,2}

(1.南京邮电大学计算机学院, 江苏 南京 210023; 2.江苏省无线传感网高技术研究重点实验室, 江苏 南京 210023)

摘要: Windows 域为用户之间的资源共享及信息交互提供统一的系统服务, 在便利内网管理的同时带来了巨大的安全隐患。近年来, 针对域控制器的各式攻击层出不穷, 实现自动化渗透能够灵活检测 Windows 域中存在的漏洞威胁, 保障办公网络安全稳定地持久运行, 其核心是高效挖掘环境内可行的攻击路径。为此, 将渗透测试过程进行强化学习建模, 通过智能体与域环境的真实交互发现漏洞组合, 进而验证有效的攻击序列; 基于主机对渗透进程的贡献差异, 减少强化学习模型中非必要的状态与动作, 优化路径选择策略, 提升实际攻击效率; 使用状态动作删减、探索策略优化的 Q 学习算法筛选最优攻击路径, 自动验证域环境中所有可能的安全隐患, 为域管理员提供防护依据。实验针对典型内网业务场景展开测试, 从生成的 13 种高效攻击路径中筛选最优路径, 通过与相关研究成果对比, 突出了所提方法在域控权限获取、主机电权获取、攻击步长、收敛性以及时间代价等方面的性能优化效果。

关键词: Windows 域; 渗透测试; 强化学习; 攻击路径

文献标志码: A

DOI: 10.11959/j.issn.2096-109x.2023057

Automated Windows domain penetration method based on reinforcement learning

ZHAN Lige^{1,2}, SHA Letian^{1,2}, XIAO Fu^{1,2}, DONG Jiankuo^{1,2}, ZHANG Pinchang^{1,2}

1.College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

2. Jiangsu Provincial Key Laboratory of Wireless Sensor Network High Technology Research, Nanjing 210023, China

Abstract: Windows domain provides a unified system service for resource sharing and information interaction among users. However, this also introduces significant security risks while facilitating intranet management. In recent years, intranet attacks targeting domain controllers have become increasingly prevalent, necessitating automated penetration testing to detect vulnerabilities and ensure the ongoing maintenance of office network operations. Then efficient identification of attack paths within the domain environment is crucial. The penetration process was first modeled using reinforcement learning, and attack paths were then discovered and verified through

收稿日期: 2022-05-21; 修回日期: 2023-03-02

通信作者: 沙乐天, ltsha@njupt.edu.cn

基金项目: 国家重点研发计划 (2018YFB0803400); 国家杰出青年科学基金 (62125203); 国家自然科学基金 (62072253)

Foundation Items: The National Key Research and Development Program of China (2018YFB0803400), The National Science Fund for Distinguished Young Scholars of China (62125203), The National Natural Science Foundation of China (62072253)

引用格式: 占力戈, 沙乐天, 肖甫, 等. 基于强化学习的自动化 Windows 域渗透方法[J]. 网络与信息安全学报, 2023, 9(2): 104-120.

Citation Format: ZHAN L G, SHA L T, XIAO F, et al. Automated Windows domain penetration method based on reinforcement learning[J]. Chinese Journal of Network and Information Security, 2023, 9(2): 104-120.

the interaction of the model with the domain environment. Furthermore, unnecessary states in the reinforcement learning model were trimmed based on the contribution differences of hosts to the penetration process, aiming to optimize the path selection strategy and improve the actual attack efficiency. The Q-learning algorithms with solution space refinement and exploration policy optimization were utilized to filter the optimal attack path. By employing this method, all security threats in the domain can be automatically verified, providing a valuable protection basis for domain administrators. Experiments were conducted on typical Windows domain scenarios, and the results show that the optimal path is selected from the thirteen efficient paths generated by the proposed method, while also providing better performance optimization in terms of domain controller intrusion, domain host intrusion, attack steps, convergence, and time cost compared to other approaches.

Keywords: Windows domain, penetration testing, reinforcement learning, attack path

0 引言

随着信息时代的发展,办公网络不断趋于复杂化、整体化,以工作组模式将每台主机进行独立管理逐渐无法满足业务需求,借助 Windows 域搭建内网环境成为主流。Windows 域是计算机网络的一种组织形式,通过名为域控制器的中央数据库实现域内的信息存储、身份认证和权限管理,保证主机执行一致的安全策略,降低管理难度^[1]。借助域网络,用户之间可以实现资源共享和信息交互,有效提高资源利用率。然而,在提供便捷的同时,Windows 域带来了巨大的安全隐患。首先,域内主机均具备与域控制器的交互权限,域控制器的位置很容易遭到暴露与攻击;其次,每个域账户可能具有多台域主机的权限,利用防护较弱的主机作为跳板,攻击者可以轻松渗透域内其他主机,当该主机中存在具有域控制器权限的域管理员凭据时,攻击者能够利用其渗透整个域。此外,环境本身存在的漏洞也可能导致高权限的丢失:域内使用特殊的认证协议(如 Kerberos^[2]、NTLM^[3])与权限策略(如委派^[4])完成身份验证和权限分配,当协议存在缺陷、策略被滥用时(如 CVE_2021_42278、CVE_2021_42287 等),攻击者可以绕过身份认证过程完成对高权限用户的凭据窃取,进而严重威胁域网络的安全。Windows 域的广泛应用和安全隐患凸显了对其进行渗透测试的必要性。

渗透测试(penetration testing)^[5]是一种发现安全威胁的常用方法,测试者通过模拟攻击行为发现网络中可能存在的脆弱性问题,较被动检测更具可信度和说服力,但需要消耗安全研究人员

大量的时间和精力,若实现面向域网络的自动化渗透测试(automated penetration testing)^[6-7],就能灵活高效地进行安全评估,减少人力和资源的浪费,其核心在于自动生成攻击路径^[8]——一种用于获取目标网络中特定资产的漏洞序列,而最优攻击路径^[9]则是攻击者最有可能使用、能以最小代价达到渗透目的的路径,实现最优攻击路径的准确预测,就可利用有限防御资源进行重点部署,切断关键入侵节点。使用强化学习解决路径发现问题是当前的研究热点,该方法通过模拟攻击者的信息搜集与动作执行过程,自动地探索、发现动作序列,并基于渗透效率对比筛选最优攻击路径,但方法在是否调用攻击载荷进行脆弱性验证上产生了分歧:调用攻击载荷能够实际验证目标网络是否存在对应漏洞,虽然路径结果的可信度高,但验证效率较低,只适合小规模网络;依据漏洞扫描结果的静态分析方法具有较高的路径发现效率,可以应用于大型网络,缺点是可行性不高,任何环节出错均可能导致整条路径失效。因此,如何在保证攻击路径真实可用的前提下提高生成效率是当前亟须解决的问题。

综上,本文提出一种基于强化学习(reinforcement learning)^[10]的自动化 Windows 域渗透方法,将渗透测试过程进行强化学习建模,通过强化学习与域环境的真实交互发现攻击路径,并使用优化的 Q 学习算法^[11]筛选最优攻击路径,所提方法能够自动验证域环境中所有可能的安全隐患,为域管理员提供防护思路,提高防御资源利用率。本文的主要贡献如下。

1) 通过强化学习与域环境的真实交互,所提

方法能够自动验证域环境内所有可行的攻击路径,并依据路径内渗透动作导致的权限增益及漏洞损耗判别其优劣,从而筛选兼具渗透效率与效果的最优路径。

2) 结合基于主权限限的状态枚举方法与基于渗透价值的求解空间优化方法,降低强化学习策略的存储难度,减少模型的非必要状态与动作,缓解状态爆炸导致的算法收敛问题,在同类实验环境下实现对相关Q学习算法33%的效率提升。

3) 将域渗透中基于用户凭据的渗透方法与传统渗透方法相结合,解决现有研究完全依赖主机漏洞进行路径发现的问题,提高所提方法在域环境内的适用性与攻击效果。

1 研究现状

根据攻击路径生成方法基于的理论不同,可以将其分为:基于攻击图模型的自动化攻击路径生成方法、基于规划模型的自动化攻击路径生成方法和基于马尔可夫决策模型的自动化攻击路径生成方法。

1.1 基于攻击图模型的自动化攻击路径生成方法

为了主动发现系统中存在的安全隐患,更清晰地分析入侵者可能采取的行动,文献[12]提出使用攻击树来表达攻击行为和攻击步骤间的依存关系,树的每个节点代表一个攻击行为或一个子任务,根节点代表攻击的最终目标,概述了入侵者从实施攻击到完成任务的全过程,其局限在于只能面向单目标进行路径发现,不适用于多目标场景。为了解决这个问题,文献[13]提出基于攻击图模型^[14]的路径发现方法,攻击图根据目标网络的脆弱性信息及潜在威胁枚举该网络中所有可能的攻击路径,缺点是生成与分析都较为复杂,很难进行人工处理^[15]。为此,文献[16]提出一种智能^[17]解决方法:首先使用多阶段漏洞分析语言(MulVAL, multi-stage vulnerability analysis language)^[18]对漏洞扫描结果进行攻击图建模,再基于Q学习算法完成路径分析,筛选出每个目标状态的最优攻击路径。然而,随着目标网络的主机与漏洞数量不断增多,攻击图规模呈指数增长,传统Q学习算法很难进行求解。为了分析大型网络中的攻击图模型,文献[19]提出基于深度强化

学习的DQN(deep Q-network)算法^[20-21]的攻击路径生成方法,利用深度神经网络^[22]拟合攻击图中的状态和动作,成功解决了状态爆炸问题。但该方法假定主机的脆弱性信息与漏洞的成功利用之间存在绝对的因果关系,缺乏真实环境下的漏洞验证。为了考虑渗透测试过程中的不确定性,文献[23]将攻击图节点的重要性与最大可达概率进行量化,以概率的形式评估目标网络的脆弱性可能带来的安全风险,使攻击路径更加贴合实际。

1.2 基于规划模型的自动化攻击路径生成方法

文献[24]提出基于确定性规划算法的攻击路径生成思路,该方法将攻击模型转化为规划领域定义语言(PDDL, planning domain definition language)^[25],用于描述网络场景的漏洞信息及利用关系,随后使用经典规划等算法进行分析,解决静态、确定、完全观测条件下的路径规划问题^[26],但其缺点是,确定性规划算法的实现基于已知目标网络的完整信息,无法适用于模型未知或存在多种状态变化的情况。为此,文献[27]将动作执行的可能结果抽象为每个动作成功的概率,利用PPDDL(probabilistic PDDL)对目标网络的基本信息与攻击动作进行建模,以概率的形式表达不同的状态转移结果。当目标网络规模不断增大时,经典计划算法的效率很难得到保证,为了提高规划模型的求解效率,文献[28-30]提出基于分层任务网络(HTN, hierarchical task network)的路径发现算法,将庞大的目标任务不断分解为更小的子任务求解。然而,以上方法只停留在理论分析,没有将生成的攻击路径用于实际渗透过程。为此,商用工具Core-Impact^[31]将计划生成器Metric-FF^[32]与渗透测试工具结合,利用后者执行前者生成的攻击路径,实现从信息搜集到漏洞利用的完整攻击过程。

1.3 基于马尔可夫决策模型的自动化攻击路径生成方法

马尔可夫决策过程(MDP, markov decision process)^[33]可以解决有限状态、动作下的路径生成问题,文献[34]使用强化学习方法求解MDP问题,通过多回合地探索、利用,得到每个状态对应的最佳漏洞,但方法需要测试者手动进行漏洞验证,且没有对问题的求解规模进行优化,因此

可行性不高。文献[35]为 MDP 构建自动化系统，其动作空间由漏洞攻击载荷组成，状态空间由漏洞利用结果组成，模型的目标是找到漏洞利用代价最小的路径序列。然而，该方法无法考虑马尔可夫过程中状态转移的不确定性，为此，文献[36]提出使用部分观测的马尔可夫决策过程（POMDP，partially observable markov decision process）^[37]进行安全分析，在 MDP 模型的基础上增加了状态转移的可能性，方法通过网络配置信息预测不同状态下漏洞利用效果的回报期望，从而对攻击动作进行优先级排序。由于 POMDP 的计算复杂性随着问题规模呈指数增长，一般只适用于求解一些小规模问题。因此，文献[38]提出了一种基于上下界的信念点优化方法，该方法使用 MDP 探索来自初始信念点的最优可达状态，并在值迭代期间进行动作选择，有效提升了求解效率。

综上可知，强化学习算法在环境中不断试错并改进自身策略的学习模式，同渗透测试持续验证漏洞、优化攻击路径的过程相似，因此被广泛应用于上述模型的求解。区别在于，攻击图模型或规划模型通过环境信息提取可能存在的安全隐患，强化学习并没有同目标环境进行真实交互，因此无法保证漏洞的可行性，而 MDP 模型通常使用漏洞载荷进行验证，能够输出真实的攻击路径，但在网络不断扩大的情况下很难保证算法的效率及收敛性。因此，已有方法依旧存在求解效率与路径可信度难以同时满足的矛盾，且没有考虑目标网络中主机的关联性，不适用于主机联系紧密、网络整体性强的域环境。为了解决上述问题，本文提出一种基于强化学习的自动化 Windows 域渗透方法，利用强化学习分析渗透工具与目标环境交互产生的攻击路径，并充分考虑域环境的策略特点和域网络的内部结构，在删除非必要状态、优化求解空间的基础上，增加了凭据利用等渗透方法，提高路径生成效率、收敛效率的同时，保证了攻击路径的有效性。

2 本文方法

2.1 背景知识

本文基于强化学习提出了自动化 Windows 域渗透方法，相关背景知识涉及 3 部分内容，包括

Windows 域渗透测试的基本流程、强化学习的理论基础以及利用强化学习进行自动化域渗透的可行性与实现思路。

2.1.1 Windows 域渗透测试

Windows 域渗透测试是指在已获得靶标（目标网络内已控制的主机）的条件下攻击域内其他主机，最终渗透域控制器进而控制整个域的过程。域渗透主要包含以下步骤。

1) 信息搜集：攻击者对目标主机的端口、服务、操作系统等信息进行探测，发现可能存在的脆弱性问题；

2) 漏洞利用：根据漏洞造成的不同结果分为两类。

① 权限获取：攻击者根据已知信息推测并使用相关漏洞获取主机权限。

② 权限提升：攻击者使用漏洞和配置不当等方式，将已获取的低权限用户提升为高权限。

3) 横向移动：攻击者成功控制一台主机后，以该主机为跳板渗透进同网络中的其他主机，从而进一步扩大对该系统的控制程度。

如图 1 所示，传统内网的渗透方式主要包括网页漏洞、提权漏洞以及主机层漏洞。攻击者通过搜集信息分析主机脆弱性，利用漏洞实现权限获取与提升，之后重复进行横向移动，控制其他内网主机。而域渗透与传统内网渗透在方法上有很大不同，集中体现在以下几点。

1) 针对性：在域渗透中，域控制器具有域内所有主机的最高权限，即域管理员权限，攻击者只需实现对域控制器的渗透即可完全控制整个域。

2) 整体性：一个域账户可能具有多台域内主机权限，利用域用户、域管理员登录时遗留在内存中的凭据，攻击者可以在获取该主机的高权限后进行凭据窃取，从而高效控制该凭据权限内的所有主机。

3) 灵活性：除传统内网的攻击方式外，结合域漏洞及域内主机、用户的关联性进行渗透具有更优的攻击效率。

由图 1 可知，渗透测试需要不断对目标网络进行信息搜集与攻击验证，其特点可类比为强化学习与环境交互的动作优化过程。本文将渗透测试过程进行强化学习建模，通过渗透工具与环境

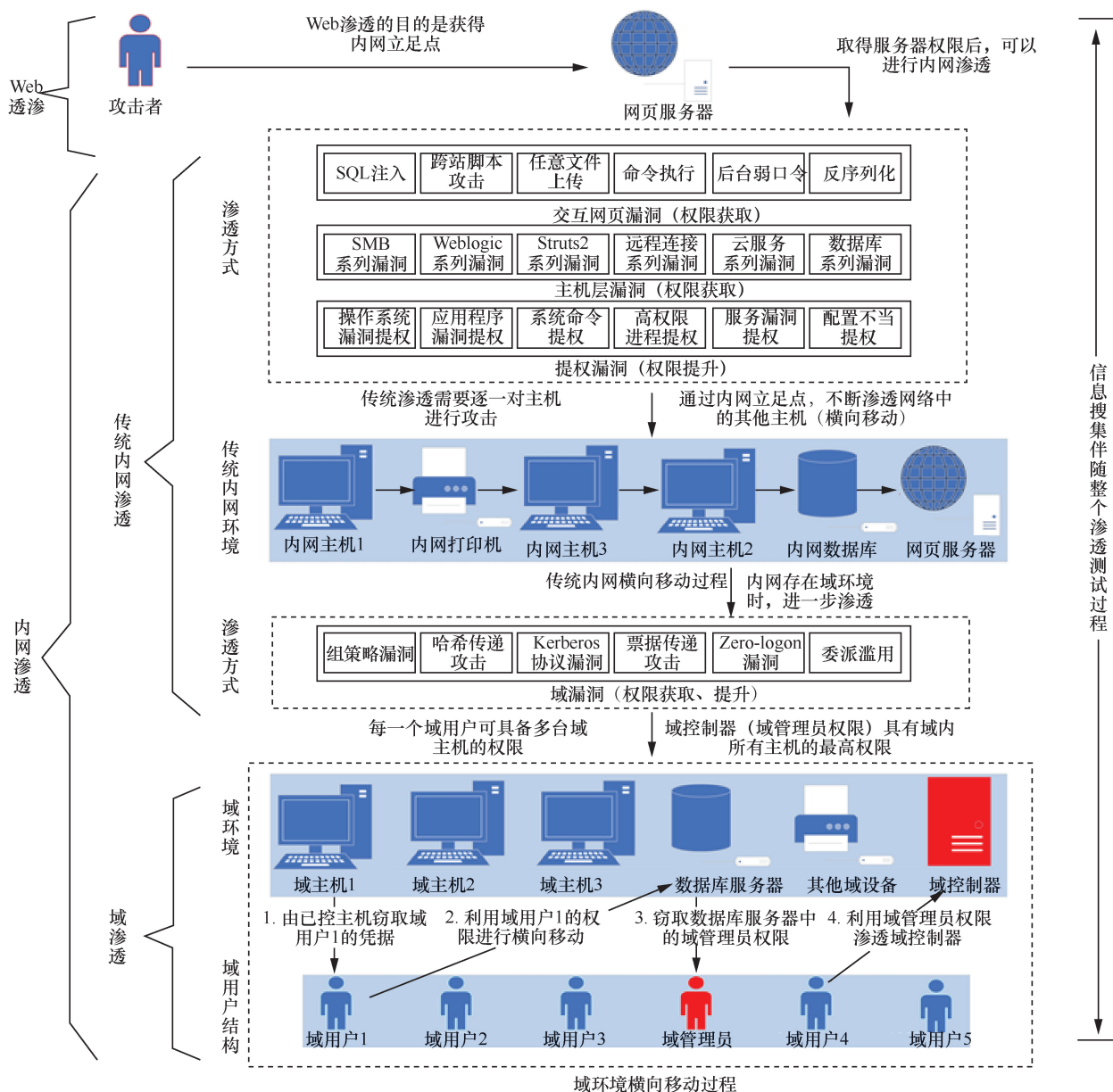


图1 渗透测试过程概述

Figure 1 Overview of the penetration testing process

的真实交互改进最优路径策略, 实现渗透测试的自动化。

2.1.2 强化学习

强化学习是指从环境状态到动作映射的学习, 学习者必须在环境中不断探索试错得到正向、反向的奖励(reward), 依据奖励判断动作的优劣。

强化学习描述为MDP五元组 (S, A, R, P, γ) 形式, 其中, 状态空间 S 包含所有可能到达的状态; 动作空间 A 为所有可能的动作集合; 奖励函数

$R(s, s', a)$ 用于衡量状态 s 下执行动作 a 后转移到下一个状态 s' 所取得的奖励($s, s' \in S, a \in A$); 状态转移 $P(s, s', a)$ 为状态 s 下执行动作 a 后转移到其他状态 s' 的概率($s, s' \in S, a \in A$); 折扣因子 $\gamma \in (0, 1)$ 体现了模型对当前奖励和未来奖励的取舍。

强化学习中, 策略 $\pi(a|s)$ 决定了当前状态 s 下智能体采取的动作 a , 学习的最终目标是优化策略 π 以最大化累积奖励。

$$E(G_t | \pi) = R_{t+1} + \gamma R_{t+2} + \dots + \gamma^{n-1} R_{t+n} \quad (1)$$

其中, G_t 代表 t 时刻之后的累积奖励, R_{t+n} 表示 $t+n$ 时刻的动作奖励。

Q 学习算法常用于求解上述策略函数, 该算法是典型的 Off-policy 强化学习方法, 将策略分为探索环境的行为策略和进行动作选择的目标策略, 并使用动作价值函数 $Q(s, a)$ 表示状态 $s(s \in S)$ 下采取动作 $a(a \in A)$ 后能够获得的累积奖励期望, 同时依据期望大小进行动作选择。上述函数被存储在名为 Q 表的矩阵中, 每次执行动作后, 表内的值都会得到更新, 直到算法收敛。

$$Q_t(s, a) \leftarrow Q_{t-1}(s, a) + a(R_t - Q_{t-1}(s, a)) \quad (2)$$

其中, a 为学习率。

从初始状态开始到目标状态结束的过程为一幕 (episode), 智能体到达目标状态后将进行下一幕探索, 当算法达到收敛时, Q 学习算法将得到最佳目标策略。

结合以上背景知识, 本文以渗透场景为强化学习的状态, 漏洞载荷为强化学习的动作, 根据前期搜集的信息, 在不同的状态下尝试可能的动作, 最终发现并验证最优攻击路径, 实现自动化渗透测试。

2.2 自动化系统构建

渗透测试的目的是以最少的攻击动作获取最多的主机权限, 针对不同动作成本及利用结果, 往往存在多条攻击路径, 筛选最优攻击序列, 有利于从入侵者角度发现高危节点, 维护网络安全。

系统的整体流程可以分为 3 个阶段: 信息搜集阶段、强化学习阶段、攻击执行阶段。在信息搜集阶段, 对常见的漏洞利用模块和域网络的信息进行搜集, 分别存放在漏洞数据库与信息数据库中。之后, 根据目标主机的脆弱性信息从漏洞数据库中匹配对应的模块, 并由强化学习策略选择当前状态下的最佳模块作为动作, 渗透工具执行该动作后的状态变化与环境奖励将用于强化学习训练, 随着与目标环境的不断交互, Q 学习算法逐步优化直至收敛, 系统将依据最优策略进行自动化渗透, 系统架构如图 2 所示。

为了使用强化学习算法求解渗透过程, 需要

对目标环境进行建模, 定义状态空间 S 、动作空间 A 、奖励函数 R 及折扣因子 γ , 并通过模块与实验环境的交互探索可能的状态转移 P , 发现实验环境内实际可以达到的状态。

2.2.1 状态空间建模

状态空间用于衡量攻击者当前对域内所有主机信息及权限的掌握程度。构造状态空间包括 3 个步骤。

1) 域内信息搜集: 信息搜集的目的是明确域网络内所有主机的基本状态及漏洞隐患。首先, 通过靶标搜集域内主机及用户信息; 其次, 探测主机的操作系统版本、开放端口及服务; 最后, 使用漏洞扫描工具发现可能存在的脆弱性信息, 并将上述所有信息存储于数据库。

2) 基于权限的状态空间枚举: 状态空间可描述为对主机所有可能的攻击序列集, 未优化的情况下, 理论上存在 $\sum_1^n n!$ 种不同情况 (n 为主机

数), 为了以最少的状态表示主机在渗透过程中的完整变化, 依据获取主机的权限将渗透状态划分为: 未获得目标主机权限 (N)、获得目标主机的域普通用户权限 (U_1)、获得目标主机本地普通用户权限 (U_2) 以及获得目标主机的本地管理员权限 (L), 由于具有域普通用户和本地普通用户权限的主机均可以进行信息搜集、横向移动等动作, 且两者的权限级别相同, 将这两种状态合并为“获得目标主机的普通用户权限 (U)”。之后, 结合步骤 1) 中搜集的信息, 以“IP-操作系统-端口: 服务-存在漏洞-已获得的权限”的形式将每台主机的信息存储于数据库, 若同时取得 U 、 L 权限时, 则表示为高权限 L , 如“172.16.253.17-Windows7-22:SSH-CVE_2020_15778-L”。其中, 仅权限状态会随着渗透过程发生改变, 因此使用“状态编号-已获得的权限”的形式枚举可能的渗透状态, 如状态“1:L-2:U-3:N-4:N-5:N”表示已获得主机 1 的本地管理员权限与主机 2 的普通用户权限, 尚未获得主机 3、4、5 的权限。此外, 将成功渗透域控制器的状态设为目标状态 G , 若主机 n 为域控制器, 则状态空间可枚举为“1:X-2:X-3:X-4:X-5:X...n-1:X-n:Y”的集合, 其中 X 取 N 、 U 、 L , Y 取 N 或 G , 而 X 、 Y 的数值为具体的状态, 如

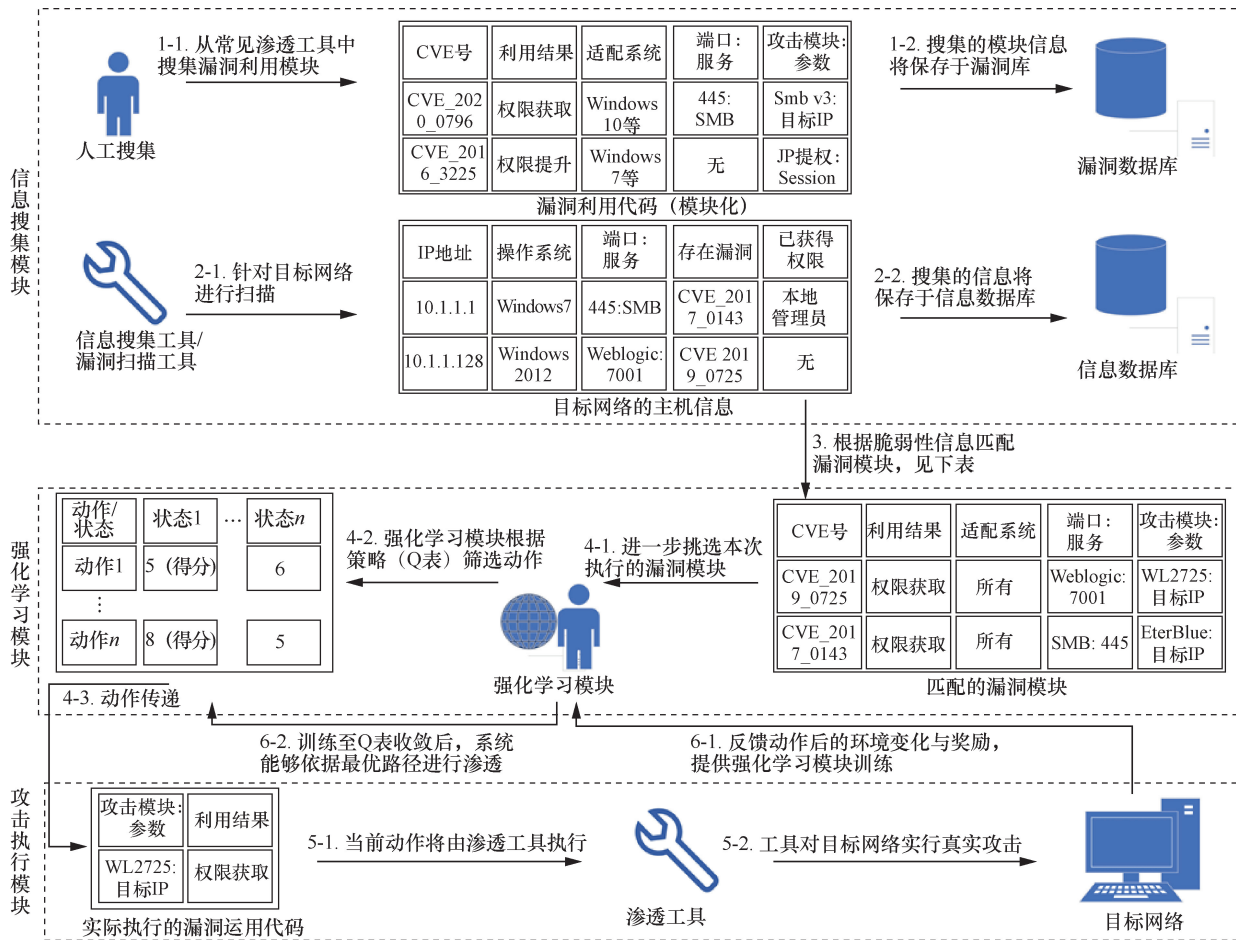


图2 系统架构
Figure 2 System architecture

“1:L-2:N-3:L-4:U-5:N...n-1:N-n:G”表示攻击者成功取得目标,且渗透域控制器前的状态为“1:L-2:N-3:L-4:U-5:N...n-1:N”。由此,将状态空间的数量由 $\sum_1^n n!$ 降至 3^n ,降低Q表的存储难度。

2.2.2 动作空间建模

动作空间代表攻击者可能使用的漏洞利用动作,其主要来源于渗透工具 Metasploit Framework (MSF)^[39]中不同的攻击模块,每个漏洞模块以“CVE号-漏洞利用结果-适配系统-端口:服务-攻击载荷及参数”的形式存储在漏洞库内,如“CVE_2019_0708-权限获取-Win7-3389:RDP-2019_0708_RDP:targetIP”。

为了提高域渗透的效率,引导智能体优化动作选择,将漏洞利用模块细分为凭据利用模块及传统漏洞模块(含权限获取动作、权限提升动作),

除了利用传统漏洞进行渗透外,还可通过已窃取域的域用户凭据进行横向移动。此外,系统会在成功渗透主机后自动添加网络路由,解决防火墙或多层网络导致的主机不可达问题。空间建模流程如图3所示。

综合以上分析,状态空间和动作空间实质上是由渗透测试过程中所需的要素组成的。前者对应渗透中的信息搜集,通过主机的基本信息与当前所取得的权限区分渗透状态;后者包含常用的漏洞载荷,对应渗透中的漏洞利用动作。通过二者结合,强化学习能够模拟攻击者从信息搜集到漏洞利用的整体渗透过程。

2.2.3 基于渗透价值的求解空间优化

为了加快强化学习的训练效率,需要尽可能删除无意义的状态与动作。在域环境中,由于夺取域控制器后可以控制任意设备,因此渗透主机主要具有两个目的(进行内网穿透或窃取未知的

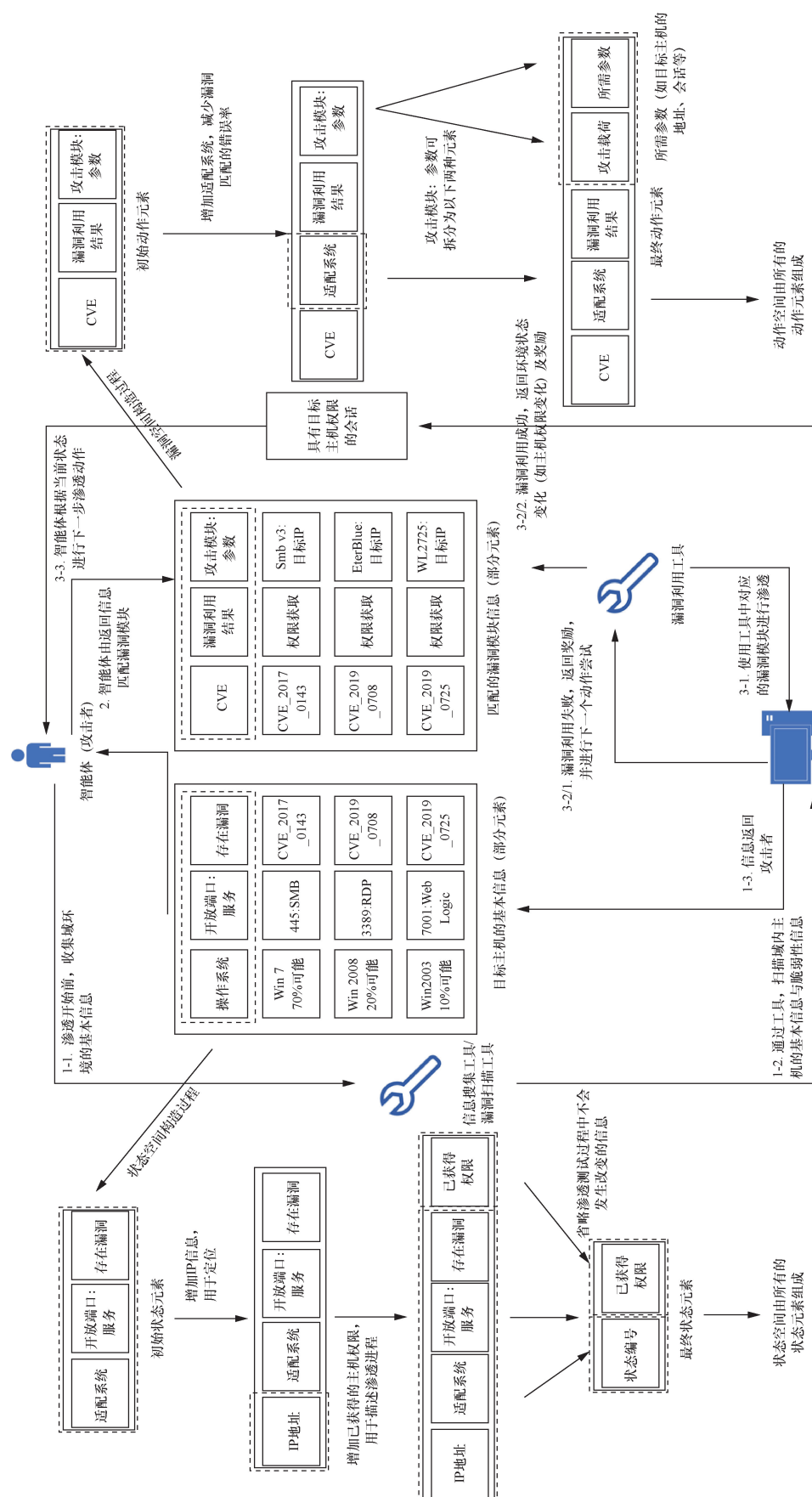


图 3 空间建模流程

域账户凭据),前者有助于突破防火墙、路由的限制,如渗透内网堡垒机;后者则用于域间的横向移动,使攻击者快速控制其权限下的主机。当一台主机不具备以上两种特性,即无法帮助进行内网穿透且不含未知凭据时,则认为其不具有渗透价值。以下通过两种情况说明如何进行冗余主机删减。情况 1:在自动化渗透中,若已获取域用户 User1 的权限,且在路径探索中发现域主机 PC1~PC3 仅含 User1 权限,同时所有主机均处于同一网络范围之内,攻击者无法通过其中某一台主机作为跳板攻击其他网段内的主机,则此时 PC1~PC3 不具备渗透价值。情况 2:若存在另一批域主机 PC4~PC6,其中含有的域用户权限仅为 User2,且未存在连接不同网段的主机,则 PC4~PC6 具有等同的渗透价值,此时控制其中任一台主机的收益与控制所有主机相同,系统将随机选取某台主机进行渗透。综上,在信息搜集阶段及强化学习训练过程中,当发现存在满足以上两类状态简化条件的主机时,将删除其所有相关状态及漏洞利用动作,更新动作空间以及状态空间,保证智能体不进行无意义的攻击尝试。

2.2.4 奖励函数生成

强化学习依赖与环境交互后的奖励做出决策。执行任意动作后,智能体都会依据奖励函数得到该动作的即时奖励,并预测动作后续的累积奖励期望。对比文献[16]、文献[19]直接使用通用漏洞评分系统(CVSS, common vulnerability scoring system)^[40]进行奖励赋值的方式,依据真实权限变化体现动作的重要性,并根据渗透测试的漏洞利用类型不同,将奖励函数分为权限获取及权限提升两类,增加了最优攻击路径的可信度与有效性。

(1) 权限获取

任何权限获取动作都可能产生权限增益 R_{PA} ,其来源于动作前后主机状态的权限得分之差,该得分依据 2.2.1 节中定义的 3 种主机状态(N 、 U 、 L)与一种目标状态 G 划分为以下 4 类。首先,未获得目标主机权限的状态得分为 0。其次,渗透域控主机、获取域管理员权限代表本次攻击任务完成,通过赋予该状态最大奖励,从而引导强化学习在满足攻击条件后立即渗透域控制

器,提高测试效率。最后,相较于普通用户权限,窃取本地管理员权限后,攻击者能直接进行凭据窃取等高危操作,因此状态得分较普通权限更高,综合上述分析与实验测试结果,将 4 类状态的得分公式定义如下。

未获得目标主机权限: $r = 0$ 。

获得目标主机的普通用户权限: $r = 8$ 。

获得目标主机的本地管理员权限: $r = 10$ 。

获得域管理员权限: $r = 1000 + r_{extra}$ 。

若某主机在漏洞利用动作前后的权限得分为 r_1 、 r_2 ,则该动作的正向奖励为其权限增益 $\Delta R = r_2 - r_1 (r_2 > r_1)$,由于域管理员权限等同于域内所有机器的最高权限,因此到达目标状态还将获得额外得分补偿 r_{extra} ,其值为将该域环境中所有主机提升至本地管理员权限获取的得分之和。例如,针对拥有主机 PC1、PC2、PC3 和域控制器的域网络,从初始状态开始,获得 PC1 主机的本地管理员权限产生的正向奖励为 $\Delta R_1 = 10 - 0 = 10$;获得 PC2 的普通用户权限产生的正向奖励为 $\Delta R_2 = 8 - 0 = 8$;此时若成功渗透域控制器,除了目标状态的得分,还将获得未达到本地管理员权限主机 PC2 (获得目标主机的普通用户权限)、PC3 (未获得目标主机权限)的得分补偿 $r_{extra} = (10 - 8) + (10 - 0) = 12$,因此,其动作的奖励为 $\Delta R_3 = 1000 + 12 = 1012$ 。

为了引导智能体选择最优的漏洞利用动作,定义 r_{cost} 为漏洞利用的动作损耗,用于衡量漏洞危害性、时效性与利用难度, r_{cost} 由漏洞等级 V_{rank} 、漏洞暴露的时间 T_{vuln} (year)、漏洞可用性 $V_{exploitable}$ 这 3 个因素组成,其值分别来自 NVD (national vulnerability database)^[41]的漏洞基础得分、披露时长及可用性评价,对于平台没有及时收录的漏洞,将依据 CVSS 计算规则对其进行评分, r_{cost} 可由下式表示。

$$r_{cost} = 10 \text{Sigmoid}(T_{vuln}) - \frac{V_{exploitable} V_{rank}}{10} \quad (3)$$

此外,SSH 远程登录和 Hash 传递攻击(PTH, pass the hash)等凭据利用动作,其本质是与主机进行正常交互而非漏洞利用,因此此类动作的损耗为 0。

综上,权限获取动作的奖励为权限增益 ΔR

与漏洞消耗 r_{cost} 之差。

$$R_{\text{PA}} = \Delta R - r_{\text{cost}} \quad (4)$$

(2) 权限提升

R_{PE} 为权限提升动作产生的奖励。若提权前后主机的权限得分为 $r1'$ 、 $r2'$ ，则权限增益 $\Delta R' = r2' - r1'$ ，由于权限提升动作的漏洞利用效果通常仅与操作系统类型相关，与漏洞等级、可用性与时间的关系较小，因此忽略其漏洞损耗。

$$R_{\text{PE}} = \Delta R' \quad (5)$$

特别地，若漏洞利用动作前后的权限增益为 0，表明此时该动作无法有效推进渗透进程，为了避免无意义动作的出现，令其动作奖励

$$R_{\text{PA}} / R_{\text{PE}} = -1 \quad (6)$$

奖励函数可以体现，有效的权限提升是取得正向奖励的前提，在利用效果相同的情况下，威胁度高、利用难度低、暴露时间短的漏洞损耗较低；相较于漏洞利用，使用域用户凭据进行渗透损耗更低，越低的漏洞损耗伴随着越高的奖励。

2.2.5 基于 Q 学习的域渗透优化算法

完成上述建模后，进一步构造实验域网络，利用强化学习与目标环境的不断交互生成最优攻击路径。系统的整体流程为：渗透开始前，将预先搜集的漏洞利用模块与扫描得到的主机信息存储在对应数据库中，之后依据主机的脆弱性信息匹配合适的漏洞模块。在强化学习阶段，首先尝试使用上述模块渗透不同主机，若成功获取主机权限，则以其为跳板进行新一轮信息搜集，发现可能遗漏的内网资产及其漏洞情况。之后，依据提权工具 Windows-Exploit-Suggester^[42]的结果筛选针对该主机的最优提权动作，同时利用 MSF 的后渗透模块获取敏感文件、日志下的账户密码及 hash 凭据，最后优化并更新因上述过程导致的状态空间与动作空间变化；一定幕数后，使用行为策略 ε -greedy 进行路径探索，即以 $1-\varepsilon$ 概率使用目标策略进行动作选择，否则随机选取动作，同时依据环境反馈进行策略优化，若发现主机符合状态删减条件，则继续进行状态简化。智能体到达目标状态后将初始化环境并进入下一幕训练，最终使基于 Q 学习的域渗透优化算法完成收敛，算法将输出经过实际验证的

最优路径。

算法 1 基于 Q 学习的域渗透优化算法

定义 ΔR 为动作产生的环境奖励；设置 α 、 γ

为算法的学习率与折扣因子；

输入 学习率 α 、折扣因子 γ ；建模后的渗透环境

输出 攻击路径

- 1) 在每一幕迭代中
- 2) 初始化实验环境
- 3) IF 当前迭代幕数 $< n$:
- 4) While(环境内存在未被控制的主机 && 获得的权限增益 > 0):
- 5) 尝试攻击动作并获取凭据
- 6) IF 目标主机符合状态删减条件:
- 7) 优化冗余状态与动作
- 8) End IF
- 9) $Q_t(s, a) \leftarrow Q_{t-1}(s, a) + \alpha(r - Q_{t-1}(s, a))$
//更新 Q 表
- 10) $s \leftarrow s'$ //更新状态
- 11) End While
- 12) Else:
- 13) While(当前状态不等于目标状态)do:
- 14) 根据 $Q(s, a)$ 表选择当前状态下的最优动作
- 15) 执行动作并获取奖励
- 16) 重复步骤 6)到步骤 10)的动作
- 17) End While
- 18) END IF
- 19) return Q-table and Result
//返回收敛后的 Q 表并输出最优路径

3 实验分析

3.1 实验环境

为了验证本文方法的有效性，本节进行了实际测试，测试平台为 Dell R70 标准机架式服务器，CPU 为 2 颗 Intel 至强 10 核 20 线程处理器，内存为 128 GB DDR4。平台通过调用商用虚拟化靶场的 OpenStack API 来模拟真实公司内部域环境，如图 4 所示，包括由 Web 服务器、业务区、后勤区等组成的域网络与办公区，以工作组模式运行若干主机。网络内使用防火墙进行区域划分，

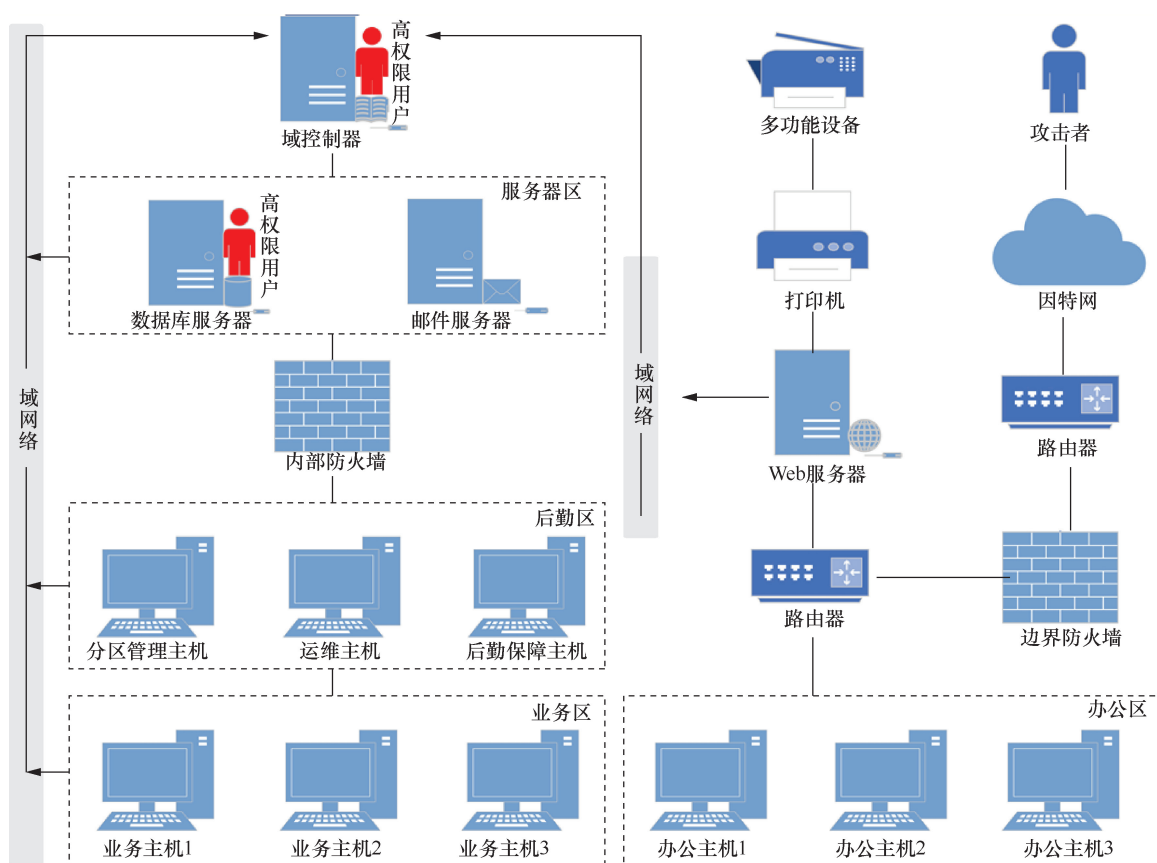


图4 实验环境

Figure 4 Experimental environment

边界防火墙允许 80 端口(HTTP、HTTPS)与 7001 端口(Weblogic)的流量通过,内部防火墙则允许 DNS、LDAP、SMTP 等服务的正常流量进入服务器区,由于维护需求,运维主机可以不受限制地与任何主机进行通信。此外,域控制器中存在高权限的域管理员账户,该账户因数据业务频繁登录数据库服务器,导致其凭据遗留在数据库服务器的内存中。

实验环境中主要含有以下安全隐患。首先,Web 服务器存在 Weblogic 反序列化漏洞(CVE_2019_2725),入侵者可以利用该漏洞获取 Web 服务器的权限,进而拿下内网靶标;运维主机、后勤保障主机以及业务主机 3 具有 SMB 漏洞(CVE_2020_0796、CVE_2017_0143、CVE_2008_4250),此类漏洞将导致主机的最高权限丢失;借助数据库服务器中存在的远程命令执行漏洞(CVE_2017_7269),攻击者可以完全控制该主机;办公区主机则均存在远程访问弱口令;

同时,域中存在 7 个用户:包含 6 个域内普通用户 User1~User6 及域管理员 Admin,其中,User1 具有数据库服务器、分区管理主机的普通权限,域管理员 Admin 具有域内所有主机的最高权限。域环境内具体的主机、漏洞信息与域用户的权限情况如表 1 所示。

3.2 环境交互

通过前期搜集获取域内主机信息及其对应漏洞模块后,本文方法将以获取主机权限为目的进行环境探索,不断积累经验、进行状态简化。

针对图 4 所示的目标环境,若攻击者由 Web 服务器开始域渗透,以贪心策略对未控制主机进行攻击尝试,直到无法控制更多主机,则将产生 $\sum_{i=1}^9 9! = 409\ 113$ 种不同渗透状态与 56 个相关动作,因此求解难度较大,而本文方法能够针对域渗透问题进行优化,减少冗余状态及动作,优化求解空间。首先,依据不同的主机权限定义状态空间,将总状态数降低为 3^9 ;其次,获取 Web

表 1 主机、漏洞信息与域用户的权限
Table 1 host, vulnerability and domain user permissions

主机：简称	编号	操作系统	漏洞：漏洞类型	主机内域用户	权限
Web 服务器：Web	1	Windows 2012	CVE_2019_2725:权限获取	User2	Web、PC5、PC6
分区管理主机：PC1	2	Windows 7	CVE_2019_0708:权限获取	User1	DB、PC1
运维主机：PC2	3	Windows 10	CVE_2020_0796: 权限获取	User3	PC2
后勤保障主机：PC3	4	Windows 7	CVE_2017_0143:权限获取	User4	PC3
业务主机 1：PC4	5	Windows 7	CVE_2018_8120: 权限提升	User5	PC4
业务主机 2：PC5	6	Windows 7	CVE_2017_0143: 权限获取	User2	Web、PC5、PC6
业务主机 3：PC6	7	Windows XP	CVE_2008_4250: 权限获取	User2	Web、PC5、PC6
数据库服务器：DB	8	Windows 2008	CVE_2017_7269:权限获取	User1	DB、PC1
			CVE_2016_3225: 权限提升	Admin	ALL
邮件服务器：MAIL	9	Windows 2008	CVE_2020_0787:权限提升	User6	MAIL
域控制器：DC	10	Windows 2008	—	Admin	ALL

服务器的本地管理员权限后，系统将窃取其中域用户 User2 的凭据，显然，由于主机 PC5、PC6 无法充当攻击跳板，符合 2.2.3 节主机删减中的情况 2，因此不需要进一步渗透含有相同凭据的二者，本文方法将删除含有 PC5、PC6 的所有状态及漏洞利用动作，将状态空间简化至 2187，减少为传统方法的 1/166。此外，依据上述优化及动态交互过程，方法删减了冗余及无法推进渗透进程的动作，并最终得到了 7 个有效攻击载

荷，从而大幅缩小了求解空间，降低了路径生成的难度。

当智能体积累了一定经验之后，此时求解空间已获得有效的删减，本文方法将采取强化学习策略进行动作选择，通过不断的动作执行、环境反馈、经验学习，最终使算法达到收敛，图 5 展示了多次状态优化后，能够达到目标的状态转移及其奖励，表 2 则展示了能够推进渗透进程的有效动作与简化后的状态。

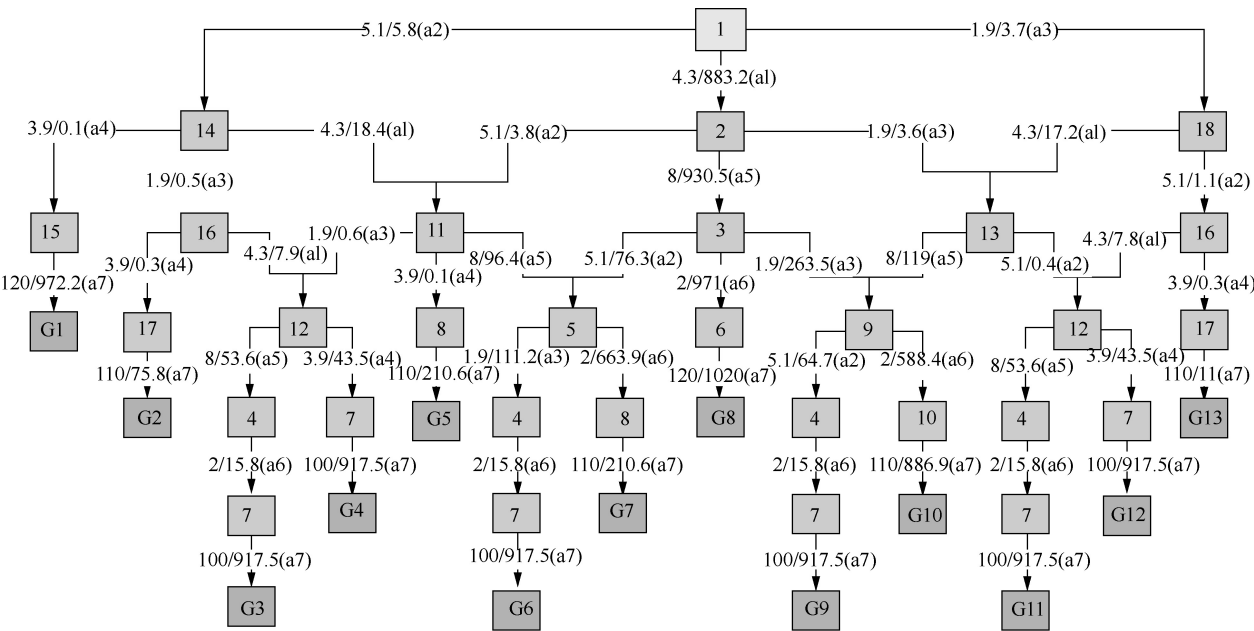


表2 有效动作及简化后的状态
Table 2 Effective actions and simplified states

编号：对应状态	动作编号：有效动作（目标）
1: 1L	a1: 漏洞利用 CVE_2019_0708(PC1)
8: 1L-2L-3L-5L	
15: 1L-3L-5L	
2: 1L-2L	a2: 漏洞利用 CVE_2020_0796(PC2)
9: 1L-2L-4L-5U	
16: 1L-3L-4L	
3: 1L-2L-5U	a3: 漏洞利用 CVE_2017_0143(PC3)
10: 1L-2L-4L-5L	
17: 1L-3L-4L-5L	
4: 1L-2L-3L-4L-5U	a4: 漏洞利用 CVE-2017-7269(DB)
11: 1L-2L-3L	
18: 1L-4L	
5: 1L-2L-3L-5U	a5: 凭据利用 User1(DB)
12: 1L-2L-3L-4L	
19: G	
6: 1L-2L-5L	a6: 漏洞利用 CVE_2016_3225(DB)
13: 1L-2L-4L	
7: 1L-2L-3L-4L-5L	
14: 1L-3L	a7: 凭据利用 Admin(DC)

图5中的方块代表此时的渗透状态，方块之间的箭头连线代表漏洞利用动作，每条连线的得分为当前动作获得的即时奖励及算法收敛后的累计奖励。在本文环境中，攻击者由Web服务器（状态1）开始域渗透，此时可能产生3种有效的转移状态，包括2、14、18，若攻击者执行动作a1并转移到状态2，它将获得的即时奖励为4.3、累积奖励为883.2，当智能体转移到目标状态G1~G13时，代表完成此幕渗透，随着强化学习与实验环境的不断交互，算法将达到收敛，并通过对比累积奖励筛选最优攻击路径。表3列举了每条到达目标状态的路径在收敛后获得的累积奖励，对比图5及表3的数据可知，路径1→2→3→6→G8在所有路径中获得了最大累积奖励3804.7，为本文方法在该实验环境下选择的最优路径。

从域渗透的角度分析，在本文环境中，由于数据库服务器具有域管理员权限的凭据，因此通过其窃取高权限凭据并渗透域控制器是一种高效

的攻击方式。该方式主要有两种实现途径：渗透分区管理主机并获取数据库服务器的普通权限（路径1→2→3），利用提权后的DB（路径1→2→3→6）窃取域管理员凭据，从而完成渗透（路径1→2→3→6→G8）；或攻击运维主机PC2，利用其作为跳板渗透DB，进而伪造域管理员身份控制整个域。相比之下，前者减少了攻击动作，且渗透手段更加隐蔽高效，其余路径均是以上两种路径的衍生，同时出现冗余动作，不是本文环境的最优路径。

表3 路径累积奖励
Table 3 Cumulative rewards for each paths

序号	攻击路径	累积奖励
1	1→14→15→G1	978.1
2	1→14→16→17→G2	82.4
3	1→2→11→12→5→7→G3	1874.5
4	1→2→11→12→7→G4	1848.6
5	1→2→11→8→G5	1097.7
6	1→2→3→4→5→7→G6	2934.5
7	1→2→3→4→8→G7	2764.5
8	1→2→3→6→G8	3804.7
9	1→2→3→9→5→7→G9	3075.2
10	1→2→3→9→10→G10	3552.5
11	1→2→13→12→5→7→G11	1874.1
12	1→2→13→12→7→G12	1848.2
13	1→18→16→17→G13	16.1

3.3 方法对比

为了验证本文方法在域渗透环境中的优越性，在图4同样的网络环境下，分别给出文献[13]方法、文献[16]方法、文献[40]方法与本文方法在渗透效率及效果、算法的收敛性与时间代价上的对比。文献[13]方法使用Q学习算法求解攻击图得到最优路径，文献[16]方法则使用DQN算法交互环境中的脆弱性信息，通过神经网络拟合路径策略并输出攻击路径，而文献[40]方法是工业界中一款开源自动化域渗透工具，主要凭借域内配置漏洞与用户凭据进行横向移动，为了提高该工具的环境适应性，为其增加了部分漏洞利用模块。

3.3.1 渗透效率及效果对比

域渗透中，能否取得域控制器权限，是评估

域网络安全风险的核心因素, 成功获取域控权限能够控制整个域网络, 若无法渗透域控, 则通过对比方法引导的自动化渗透测试能够获取的主机权限数 (N_h) 反映其渗透效果; 攻击路径的步长效率为渗透主机数 N_h 与其过程中本文方法执行动作总数 N_a 的比值 ($\frac{N_h}{N_a}$), 体现了本文方法的

攻击效率和暴露风险。

图6给出了4种不同方法的步长效率比与权限获取情况, 其输出结果为10次测试的平均值。比较前者可以发现, 本文方法的攻击路径更短、渗透效率更高, 这是由于本文方法将传统渗透手段与域渗透手段相结合: 先通过攻击主机获得域用户凭据, 后利用已知凭据在主机间快速进行横向移动。而文献[13]、文献[16]方法仅利用传统漏洞进行渗透, 未能发挥域渗透思想在域环境中的效率优势; 文献[40]方法虽然能够结合上述思想, 但方法本身不具备学习能力, 无法通过经验优化渗透路径效果, 导致步长效率不高。

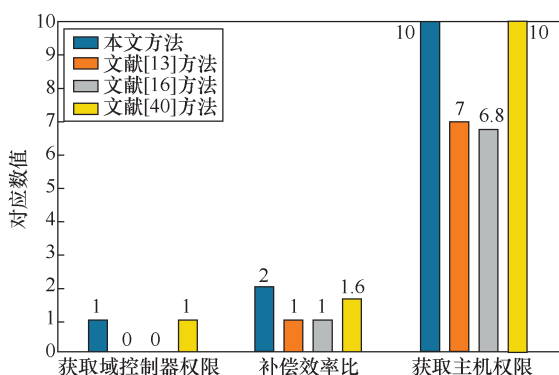


图6 可用性 & 效率对比

Figure 6 Comparison of availability and efficiency

此外, 文献[13]、文献[16]方法并未成功获得域控制器权限, 是因为在如图4所示的实验环境中, 域控制器并不存在直接漏洞, 但上述方法依赖扫描工具分析主机漏洞生成攻击路径, 当目标主机中不存在漏洞或者扫描工具漏报、误报时, 渗透将无法正确进行, 而本文方法基于漏洞利用模块与环境的真实交互, 同时利用域内高权限用户作为跳板, 对不存在漏洞的主机进行绕过或间接控制, 前者保证了攻击路径的可用性, 后者则提高了路径发现能力和渗透成功率。借助上述思想, 本文方法利用环境内 User1 及域管理员凭据

绕过防护稳固的邮件服务器及业务主机2进行横向移动, 通过获取域控制器权限渗透整个域网络, 因此取得了比文献[13]、文献[16]方法更好的攻击结果。综合考虑渗透效果和效率, 本文方法优于以上3种方法。

3.3.2 收敛性对比

攻击路径的收敛性体现了自动化算法的有效性, 是判别方法能否适用于当前环境的重要指标, 同时从侧面反映了算法效率。

图7展示了本文方法、文献[13] (该方法仅能针对每台存在漏洞的主机单独生成攻击路径, 在本文实验环境中存在6种不同情况) 和文献[16]方法的收敛性对比。训练开始后, 文献[13]方法的曲线明显上升, 均在186幕左右达到收敛。而本文方法由于初期的随机动作探索, 前几幕内得分较低, 但在34幕之后增长迅速, 并于248幕收敛。文献[16]方法于300幕达到收敛。考虑到文献[13]方法仅针对单一主机进行路径分析, 而本文方法的生成结果则能够渗透进域内的所有9台主机。结合以上实验数据可以说明, 本文方法在此环境下的收敛效果优于文献[13]方法与文献[16]方法, 这是由于本文方法的奖励函数能够有效发现真实可用的渗透动作组成攻击路径, 并依据函数中的漏洞损耗筛选最优动作, 提升了本文方法在实际环境下的适应性和收敛性, 此外, 依据渗透价值进行冗余主机删减在很大程度上降低了问题的求解难度, 进一步提高了所提方法的收敛速率。

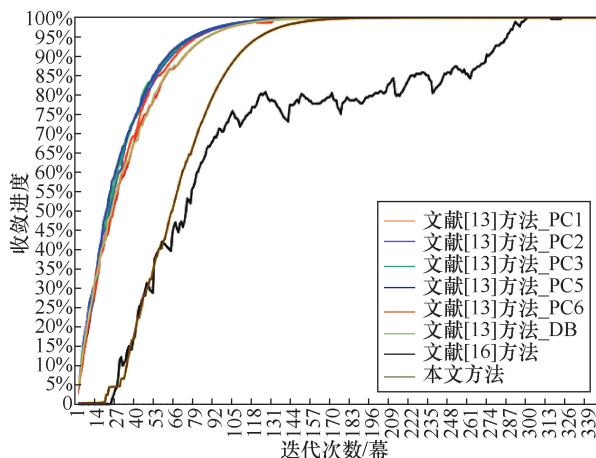


图7 收敛性对比

Figure 7 Comparison of convergence

3.3.3 时间代价对比

为了评价算法的实际执行效率,实验测试了主机数目为4、5、8、10时本文方法的总体时间代价(主机数为10即图4实验环境),包括动作执行时间、环境反馈耗时以及算法优化时间,由于文献[13]和文献[16]方法通过学习目标环境的脆弱性信息生成攻击路径,并未与环境发生真实交互,仅存在算法优化时间,为了更直观地计算及对比时间代价,在两种方法中增加了相关动作的平均时延计算,该值为目标环境内执行渗透动作所需的平均耗时。同时,由于文献[13]方法将每台主机作为目标进行独立路径求解,其时间代价为各目标耗时的总和。3种方法的实际时间损耗对比如图8所示。

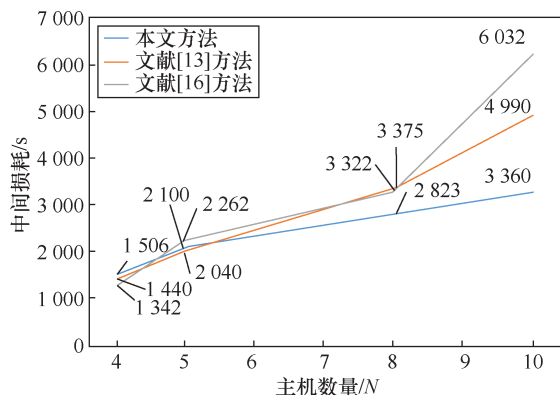


图8 实际时间损耗对比
Figure 8 Comparison of actual time loss

从上述实验结果来看,当主机数量较小时,3种方法的时间损耗并未产生较大差距,随着主机数量不断增加,其耗时开始快速增长,这是由于强化学习需要多次遍历可能的状态、动作二元组,主机数量增长同时导致渗透状态及漏洞利用动作随之增加,使得整体求解难度大幅度提高。当主机的数量超过8时,文献[13]以及文献[16]方法的总体耗时超过本文方法33%以上,由于本文方法进行主机删减优化了求解空间,使其在一定规模的网络内依旧保持着良好的求解效率;而攻击图规模和冗余节点随着主机数量增加呈指数增长,因此求解效率较低,虽然文献[13]方法通过删除非漏洞利用节点等手段简化了攻击步骤,但在有限效率优化的同时牺牲了路径的完整性;文献[16]方法需手动标注某些关键节点的奖励,

并将原始攻击图的全部可能路径输入深度神经网络,导致时间损耗显著提高。

4 结束语

为了维护Windows域网络安全,预知入侵者可能发起的攻击,为域管理员提供安全防护策略,本文提出了一种将强化学习应用于Windows域环境的安全评估方法。首先,对目标网络的脆弱性信息与漏洞工具的利用模块进行强化学习建模,构造奖励函数评价漏洞利用动作的实际价值;其次,定义域渗透中的状态简化方法提升路径求解效率,并通过强化学习与实验环境的不断交互对最优攻击路径进行验证;最后,从渗透效果与效率、收敛性及时间代价3个方面同其他攻击路径生成方法进行对比,突出本文方法在Windows域渗透中的有效性和优越性。在实际渗透中,获取主机权限后通常伴随提权操作,实现权限获取后的自动化提权,能够在提升效率的同时更加贴合渗透流程,下一步将对此展开研究,优化自动化域渗透方法。

参考文献:

- [1] 尹圣超. Windows域攻防关键技术研究[D]. 西安: 西安电子科技大学, 2021.
YIN Z C. Research on key technologies of attack and defense of Windows domain[D]. Xi'an: Xidian University, 2021.
- [2] SHEBLI H, BEHESHTI B D. A study on penetration testing process and tools[C]//2018 IEEE Long Island Systems, Applications and Technology Conference (LISAT), 2018:1-7.
- [3] STEFINKO Y, PISKOZUB A, BANAKH R. Manual and automated penetration testing, benefits and drawbacks, modern tendency[C]//2016 13th International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science(TCSET), 2016: 488-491.
- [4] ALMUBAIRIK N A, WILLS G. Automated penetration testing based on a threat model[C]//2016 11th International Conference for Internet Technology and Secured Transactions (ICITST). 2016: 413-414.
- [5] 王硕, 汤光明, 寇广,等. 基于因果知识网络的攻击路径预测方法[J]. 通信学报, 2016, 37(10): 188-198.
WANG S, TANG G M, KOU G, et al. Attack path prediction method based on causal knowledge net[J]. Journal on Communications, 2016, 37(10):188-198.
- [6] 王硕, 王建华, 汤光明, 等. 一种智能高效的最优渗透路径生成方法[J]. 计算机研究与发展, 2019, 56(5):929-941.

- WANG S, WANG J H, TANG G M, et al. Intelligent and efficient method for optimal penetration path generation[J]. *Journal of Computer Research and Development*, 2019, 56(5): 929-941.
- [7] 高阳, 陈世福, 陆鑫. 强化学习研究综述[J]. *自动化学报*, 2004, 30(1): 86-100.
- GAO Y, CHEN S F, LU X. Research on reinforcement learning technology: a review[J]. *ACTA Automatica Sinica*, 2004, 30(1): 86-100.
- [8] WATKINS C J C H, DAYAN P. Q-learning[J]. *Machine Learning*, 1992, 8(3-4): 279-292.
- [9] SCHNEIER B. Attack trees-modeling security threats[J]. *Drdobbs Journal*, 1999, 24(12): 21-29.
- [10] SHEYNER O, HAINES J, JHA S, et al. Automated generation and analysis of attack graphs[C]//*Proceedings 2002 IEEE Symposium on Security and Privacy*. 2002:273-284.
- [11] 叶子维, 郭渊博, 王宸东,等. 攻击图技术应用研究综述[J]. *通信学报*, 2017, 38(11): 121-132.
- YE Z W, GUO Y B, WANG C D, et al. Survey on application of attack graph technology[J]. *Journal on Communications*, 2017, 38(11): 121-132.
- [12] 陈锋, 张怡, 苏金树,等. 攻击图的形式化分析[J]. *软件学报*, 2010, 21(4): 838-848.
- CHEN F, ZHANG Y, SU J S, et al. Two formal analyses of attack graphs[J]. *Journal of Software*, 2010, 21(4):838-848.
- [13] YOUSEFI M, MTETWA N, Zhang Y, et al. A reinforcement learning approach for attack graph analysis[C]//2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference on Big Data Science And Engineering (TrustCom/BigDataSE). 2018: 212-217.
- [14] 张蕾, 崔勇, 刘静,等. 机器学习在网络空间安全研究中的应用[J]. *计算机学报*, 2018, 41(9): 1943-1975.
- ZHANG L, CUI Y, LIU J, et al. Application of machine learning in cyberspace security research[J]. *Chinese Journal of Computers*, 2018, 41(9): 1943-1975.
- [15] OU X, GOVINDAVAJHALA S, APPEL A W. MulVAL: a logic-based network security analyzer[C]//*USENIX Security Symposium*. 2005:113-128.
- [16] 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). 2020: 2-10.
- [17] 刘全, 翟建伟, 章宗长,等. 深度强化学习综述[J]. *计算机学报*, 2018, 41(1): 1-27.
- LIU Q, ZHAI J W, ZHANG Z C, et al. A survey on deep reinforcement learning[J]. *Chinese Journal of Computers*, 2018, 41(1): 1-27.
- [18] MNIH V, KAVUKCUOGLU K, SILVER D, et al. Human-level control through deep reinforcement learning[J]. *Nature*, 2015, 518(7540): 529-533.
- [19] 周飞燕, 金林鹏, 董军. 卷积神经网络研究综述[J]. *计算机学报*, 2017, 40(6): 1229-1251.
- ZHOU F Y, JIN L P, DONG J. Review of convolutional neural network[J]. *Chinese Journal of Computers*, 2017, 40(6): 1229-1251.
- [20] LI Y, LI X. Research on multi-Target network security assessment with attack graph expert system model[J]. *Scientific Programming*, 2021, (3): 1-11.
- [21] OBES J L, SARRAUTE C, RICHARTE G. Attack planning in the real world[C]//*Working Notes for the 2010 AAAI Workshop on Intelligent Security (SecArt)*. 2010: 10.
- [22] HASLUM P, LIPOVETZKY N, MAGAZZENI D, et al. An introduction to the planning domain definition language[J].*Synthesis Lectures on Artificial Intelligence and Machine Learning*, 2019, 13(2): 1-187.
- [23] 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 and Information Technology*, 2020, 42(9): 2095-2107.
- [24] SARRAUTE C, RICHARTE G, LUCÁNGELI O J. An algorithm to find optimal attack paths in nondeterministic scenarios[C]//*Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence*. 2011: 71-80.
- [25] SOHN S, OH J, LEE H. Hierarchical reinforcement learning for zero-shot generalization with subtask dependencies[C]//*The 32nd Conference on Neural Information Processing Systems*. 2018: 7156-7166.
- [26] DE SILVA L, PADGHAM L, SARDINA S. HTN-like solutions for classical planning problems: an application to BDI agent systems[J]. *Theoretical Computer Science*, 2019, 763: 12-37.
- [27] MOHR F, WEVER M, HÜLLERMEIER E. ML-plan: automated machine learning via hierarchical planning[J]. *Machine Learning*, 2018, 107(8-10): 1495-1515.
- [28] Core impact[J]. *SC Magazine: The International Journal of Computer Security*, 2010(2): 21.
- [29] HOFFMANN J. The metric-FF planning system: translating “ignoring delete lists” to numeric state variables[J]. *Journal of Artificial Intelligence Research*, 2011, 20:291-341.
- [30] 范长杰. 基于马尔可夫决策理论的规划问题的研究[D]. 合肥: 中国科学技术大学, 2008.
- FAN C J. Research on planning based on Markov decision theory[D]. Hefei: University of Science and Technology of China, 2008.
- [31] 赵海妮, 焦健. 基于强化学习的渗透路径推荐模型[J]. *计算机应用*, 2022: 1-7.
- ZHAO H N, JIAO J. Infiltration path recommendation model based on reinforcement learning[J]. *Journal of Computer Applications*, 2022: 1-7.
- [32] DURKOTA K, LISÝ V. Computing optimal policies for attack graphs with action failures and costs[C]//*The 7th European Starting AI Researcher Symposium*. 2014: 101-110.
- [33] SHMARYAHU D, SHANI G, HOFFMANN J, et al. Constructing plan trees for simulated penetration testing[C]//*The 26th International*

Conference on Automated Planning and Scheduling. 2016:121.

- [34] Kaelbling L P, Littman M L, Cassandra A R. Planning and acting in partially observable stochastic domains[J]. Artificial Intelligence, 1998, 101(1-2):99-134.
- [35] Liu B B, Kang Y, Jiang X F, et al. A fast approximation method for partially observable Markov decision processes[J]. Journal of Systems Science and Complexity, 2018, 31(6): 1423-1436.
- [36] Liu B B, Kang Y, Jiang X F, et al. A fast approximation method for partially observable Markov decision processes[J]. Journal of Systems Science and Complexity, 2018, 31(6): 1423-1436.
- [37] 王作广,魏强,刘雯雯.基于攻击树与 CVSS 的工业控制系统风险量化评估[J].计算机应用研究,2016,33(12):3785-3790.
WANG ZG, WEI Q, LIU W W. Quantitative risk assessment of industrial control systems based on attack trees and CVSS[J]. Application Research of Computers, 2016,33(12):3785-3790
- [38] 刘奇旭,张翀斌,张玉清,等.安全漏洞等级划分关键技术研究[J].通信学报,2012,33(S1):79-87.
LIU Q X, ZHANG C B, ZHANG Y Q, et al. Research on the key technology of security vulnerability threat classification[J]. Journal on Communications, 2012,33(S1):79-87.
- [39] AonCyberLabs. Windows-exploit-suggester[EB].
- [40] byt3bl33d3r. DeathStar[EB].

[作者简介]



张力戈(1996-),男,福建南平人,南京邮电大学硕士生,主要研究方向为网络攻防、渗透测试。



沙乐天(1985-),男,江苏徐州人,博士,南京邮电大学副教授,主要研究方向为网络安全、物联网攻防。



肖甫(1980-),男,湖南邵阳人,博士,南京邮电大学教授、博士生导师,主要研究方向为传感网、物联网。



董建阔(1992-),男,河北邢台人,博士,南京邮电大学讲师,主要研究方向为密码工程、高性能并行计算。



张品昌(1985-),男,安徽临泉人,博士,南京邮电大学讲师,主要研究方向为网络安全、物理层认证及卫星互联网安全。