软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2024,35(5):2268−2288 [doi: 10.13328/j.cnki.jos.007038]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

自动化渗透测试技术研究综述*

陈 可, 鲁 辉, 方滨兴, 孙彦斌, 苏 申, 田志宏

(广州大学 网络空间安全学院, 广东 广州 510555)

通信作者: 鲁辉, E-mail: luhui@gzhu.edu.cn; 田志宏, E-mail: tianzhihong@gzhu.edu.cn

摘 要: 渗透测试是发现重要网络信息系统弱点并进而保护网络安全的重要手段. 传统的渗透测试深度依赖人工, 并且对测试人员的技术要求很高, 从而限制了普及的深度和广度. 自动化渗透测试通过将人工智能技术引入渗透测试全过程, 在极大地解决对人工的重度依赖基础上降低了渗透测试技术门槛. 自动化渗透测试主要可分为基于模型和基于规则的自动渗透测试. 二者的研究各有侧重, 前者是指利用模型算法模拟黑客攻击, 研究重点是攻击场景感知和攻击决策模型; 后者则聚焦于攻击规则和攻击场景如何高效适配等方面. 主要从攻击场景建模、渗透测试建模和决策推理模型等 3 个环节深入分析相关自动化渗透测试实现原理, 最后从攻防对抗、漏洞组合利用等维度探讨自动化渗透的未来发展方向.

关键词: 自动化渗透测试; 攻击图; 强化学习; BDI-Agent

中图法分类号: TP311

中文引用格式: 陈可, 鲁辉, 方滨兴, 孙彦斌, 苏申, 田志宏. 自动化渗透测试技术研究综述. 软件学报, 2024, 35(5): 2268–2288. http://www.jos.org.cn/1000-9825/7038.htm

英文引用格式: Chen K, Lu H, Fang BX, Sun YB, Su S, Tian ZH. Survey on Automated Penetration Testing Technology Research. Ruan Jian Xue Bao/Journal of Software, 2024, 35(5): 2268–2288 (in Chinese). http://www.jos.org.cn/1000-9825/7038.htm

Survey on Automated Penetration Testing Technology Research

CHEN Ke, LU Hui, FANG Bin-Xing, SUN Yan-Bin, SU Shen, TIAN Zhi-Hong

(Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou 510555, China)

Abstract: Penetration testing is an important means to discover the weaknesses of significant network information systems and protect network security. Traditional penetration testing relies heavily on manual labor and has high technical requirements for testers, limiting the popularization depth and breadth. By introducing artificial intelligence technology into the whole penetration testing process, automated penetration testing lowers the technical threshold of penetration testing based on greatly solving the problem of heavy dependence on manual labor. Automated penetration testing can be mainly divided into model-based and rule-based automated penetration testing, and the research of the two has their respective focuses. The former utilizes model algorithms to simulate hacker attacks with attention paid to attack scene perception and attack decision-making models. The latter concentrates on how to efficiently adapt attack rules and attack scenarios. This study mainly analyzes the implementation principles of automated penetration testing from three aspects of attack scenario modeling, penetration testing modeling, and decision-making reasoning model. Finally, the future development direction of automated penetration is explored from the dimensions of attack-defense confrontation and vulnerability combination utilization.

Key words: automated penetration testing; attack graph; reinforcement learning; BDI-Agent

随着网络技术深入发展, 网络安全事件在世界各地频发, 诸如数据泄漏、勒索软件、黑客攻击等层出不穷, 其所造成的经济损失也同步显著增长. 2022 年全球网络安全态势仍然处于高位运行状态, Splunk 对 1 200 余名安全领导进行了调查, 有 49% 表示他们在过去两年存在数据泄露问题, 高于一年前的 39%[1]. Positive Technologies 发

* 基金项目: 国家自然科学基金 (U20B2046); 广东省高校创新团队项目 (2020KCXTD007); 广州市高校创新团队项目 (202032854)
收稿时间: 2023-03-27; 修改时间: 2023-05-22; 采用时间: 2023-08-07; jos 在线出版时间: 2023-12-27
CNKI 网络首发时间: 2023-12-29

布的 2022 年第 2 季度网络安全威胁研究报告[2]以及 NETSCOUT 的 2022 年上半年威胁情报报告[3]中分别指出: 有针对目标的攻击占据攻击总数的 71%, 其中针对个人的攻击占比 17%, 针对组织的攻击占比 28%, 相比第 1 季度上升 6 个百分点, 几乎 1/3 的攻击涉及勒索软件. 网络犯罪分子在 2022 年上半年发动了 6019888 次分布式拒绝服务攻击, 最大攻击带宽高达 957.9 Gb/s. 中国信息安全测评中心在《2022 年上半年网络安全漏洞态势观察》中也指出[4], 漏洞数量增长速度创新高, 0day 漏洞利用形势严峻, 漏洞实战化趋势明显, 网络安全威胁持续加剧.

世界各国为应对网络攻击带来的威胁出台了一系列法案. 美国发布《2021 财年国防授权法案》[5]、《临时国家安全战略纲要》[6]、《改善国家网络安全行政令》[7]等, 将网络空间安全作为重中之重, 致力于加强网络空间安全能力、就绪度及弹性; 欧盟发布《欧盟数字十年网络安全战略》[8]等数字政策; 网络空间安全事关国家安全, 作为重要国策, 我国不断强化网络安全顶层设计、保障体系、能力建设以及网络安全法律法规体系, 2016 年 11 月 7 日通过《中华人民共和国网络安全法》[9], 并于 2021 年 8 月出台《关键信息基础设施安全保护条例》[10], 这是我国首部专门针对关键信息基础设施安全保护工作的行政法规, 为开展关键信息基础设施安全保护工作提供了基本依据.

传统网络安全防御手段以识别并阻断网络攻击为核心, 力求拒威胁于内网之外, 但随着高隐蔽未知威胁的出现和演进, 越来越多的研究者相信网络攻击难以避免. 2015 年, 美国首任网军司令亚历山大将军就曾在中国互联安全大会上语出惊人: 世界上只有两种系统, 一种是已知被攻破的系统, 一种是已经被攻破但自己还不知道的系统.

安全源自未雨绸缪, 渗透测试技术作为主动探测网络漏洞、实现防范网络攻击的重要手段, 可及时发现目标网络系统的脆弱性并针对性修复加固[11]. 全球渗透测试市场规模预计将从 2020 年的 17 亿美元增长到 2025 年的 45 亿美元. 我国自 2016 年起就开始组织国家级护网行动, 各重大基础设施单位如电网, 金融行业等也常年组织内部护网活动, 招募白帽黑客组成红队对实际业务系统进行渗透测试, 以接近实战的形式最大程度上发现安全问题. 当前渗透测试主要依靠人工, 对测试人员的技术要求很高且花销巨大. 为减少渗透测试中的人工重复性劳动, 提高测试效率, 基于人工智能等相关技术的自动化渗透测试方法正逐步成为学术界和工业界的研究热点, 自动渗透测试与手动渗透测试的主要特点和区别如表 1[12].

表 1 自动渗透测试与手动渗透测试

分类	自动渗透测试	手动渗透测试
漏洞库	维护渗透载荷库	渗透测试人员在线搜索
痕迹清理	痕迹清理自动化	人工痕迹清理
技术要求	具有基本的渗透知识背景	具有专家级的渗透技术
活动日志	自动记录行为	人工记录行为
决策模型	智能模型驱动渗透测试	渗透测试人员经验

早期的自动化渗透测试是基于规则实现, 具体是将某个战术的一系列渗透动作抽象化, 并形成规则集成进自动执行的渗透工具, 所适配的场景较为单一, 诸如常见的 Nmap[13]、Nessus[14]等. 我们以“自动渗透”“渗透测试”“红队”等为关键词, 检索 WoS 数据库并筛选近 5 年的相关研究成果, 构建如后文图 1 所示的热点词图谱. 从中不难看出, 随着人工智能技术的不断成熟, 自动化渗透测试已步入模型化时代, 通过将渗透测试建模为马尔可夫决策过程和部分可观察马尔可夫决策过程, 并使用的强化学习进行自主决策, 近年来涌现出较多研究成果.

本文聚焦于自动化渗透的发展脉络和技术演进, 依据决策方式把自动化渗透技术划分为基于规则和基于模型的两个类别. 第 1 节介绍渗透测试技术的相关标准和规范. 第 2 节和第 3 节分别对基于规则和基于模型的自动化渗透的主要研究工作及代表性成果进行详细阐述, 分析相关技术路线及优势. 最后参照自动渗透技术需求, 从渗透信息提取以漏洞组合利用等方面, 进行总结和展望.

1 渗透测试技术相关标准和规范

渗透测试过程具有标准的步骤和规范, 最早是由受雇于美国五角大楼的 James 提出的时分系统安全性测试流

程[11], 具体包括: (1) 发现可利用漏洞; (2) 围绕它设计渗透计划; (3) 渗透测试; (4) 接管正在使用的线路; (5) 执行渗透计划; (6) 利用输入进行信息检索. 此基础上, 逐渐形成了渗透测试标准, 常见的渗透测试标准包括: 开源安全测试方法论 (open source security testing methodology manual, OSSTMM)[15]、信息系统安全评估框架 (information systems security assessment framework, ISSAF)[16]、渗透测试执行标准 (penetration testing execution standard, PTES)[17], 其关系如图 2 所示.

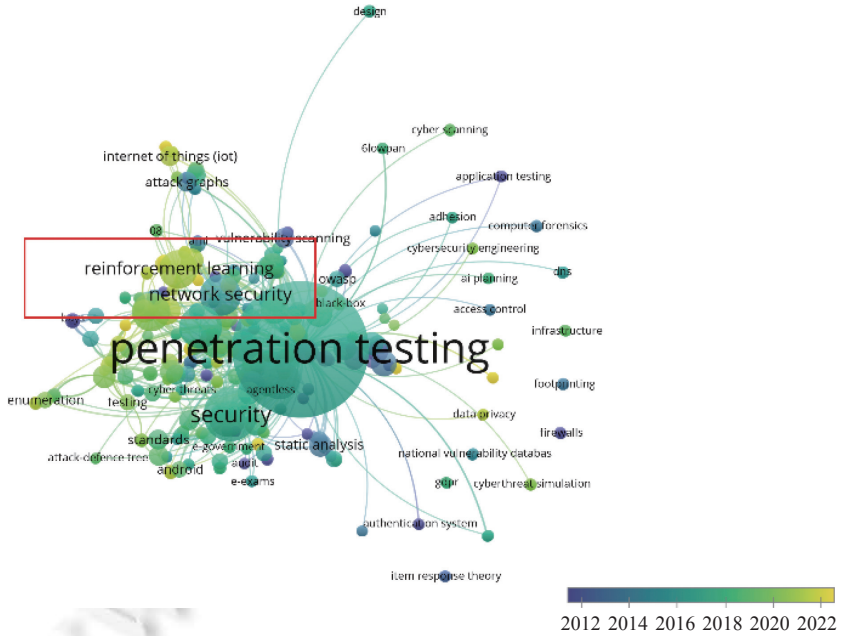


图 1 自动化渗透测试技术研究热点关键词知识图谱

图 2 渗透测试技术标准与规范

OSSTMM 是由 Pete Herzog 创建, 检查访问控制安全、流程安全、数据控制、物理位置、周界防护、安全意识水平、信任关系、反欺诈控制等诸多过程, 全面评估被测目标的安全性. OSSTMM 强调测试目标和测试方法, 注重在测试前、测试中、测试后应采用的相应策略, 并引入 RAV (risk assessment value) 分析测试结果, 进而量化评估当前安全状态, 输出安全测试报告, 方便安全管理人员理解测试结果.

ISSAF 是一个开源的安全性测试和分析框架. 相较于 OSSTMM, ISSAF 主要测试当前控制措施中的严重漏洞, 因此在保障系统安全方面意义重大. 应用 ISSAF 框架可以把精力放在特定目标上, 如路由器、交换机、防火墙、入侵检测系统、虚拟专用网络、操作系统、Web 应用服务器、数据库等, 帮助安全管理人员了解当前边界防御体系的现有风险, 并可指出可能影响业务完整性的安全弱点. ISSAF 的技术评估基准十分全面, 可用于测试各种技术和不同流程. 但其缺点也比较明显, 即若要跟上评估领域的技术变化速度, 该框架就需要频繁同步更新, 相对而言, OSSTMM 受技术更新影响的幅度略小, 同时结合 OSSTMM 和 ISSAF 两种框架可更好地评估企业环境的安全状况.

OSSTMM 和 ISSAF 都是宏观的测试框架, 并未包含渗透测试各阶段行为的具体标准. 作为补充, PTES 将标准的渗透测试分为 7 个阶段: 前期交互阶段、情报搜集阶段、威胁建模阶段、漏洞分析阶段、漏洞利用阶段、后

渗透阶段、报告阶段, 并详细介绍每个阶段的步骤. PTES 的内容更加详细, 涵盖渗透测试的技术部分和其他重要方面, 如蔓延范围、报告以及渗透人员保护自身的方法等, 同时介绍了多种测试任务的具体方法和所需技术.

显然, 上述框架与真实的黑客攻击仍然存在差距, 为更有效地建模网络渗透, MITRE 公司基于已知 APT 事件, 于 2013 年推出 ATT&CK 矩阵[18], 描述已发现的 APT 中各阶段及所使用的技术, 鉴于 ATT&CK 强大的实践指导意义, 一经推出, 就被学术界和工业界热捧, 基于 ATT&CK 矩阵的安全渗透研究和自动化渗透研究成为新一轮的研究热点.

传统渗透测试需要以给定的安全测试方法论和安全测试执行标准为规范, 利用安全知识库以及专家的渗透经验实现具体测试. 随着人工智能技术的发展, 为使渗透测试更科学, 更具说服力, AI 技术越来越多地出现在渗透测试中. 2005 年, AI 规划首次应用于渗透测试[19], 其后自动化建模被陆续应用于渗透, 使单个“渗透行为”对应于已知的软件漏洞利用[20], 经过多年发展, 自动渗透系统更加智能化. 自 2003 年至今的研究成果统计如图 3 所示, 相关成果近年来逐年增多, 有越来越多的研究者参与此方向的研究.

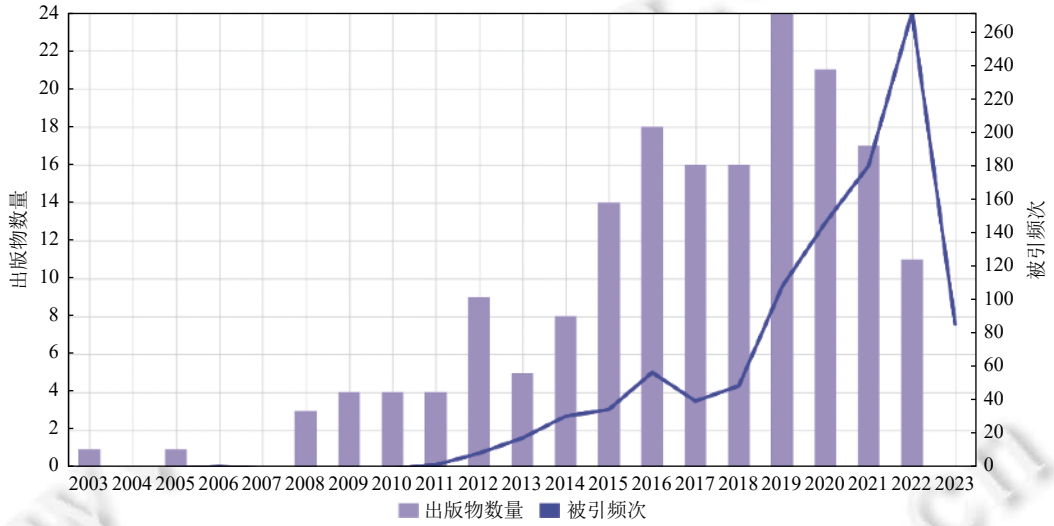


图 3 2003 年至今自动化渗透领域的出版物及被引频次

目前自动化渗透系统距离替代人实现黑客攻击的目标存在较大差距, 需深入研究和待解决的问题还有很多, 例如渗透环境建模、渗透计划建模以及渗透学习算法等. 我们基于以往研究工作[21], 将渗透测试依据层次进行划分, 归纳如图 4 所示.

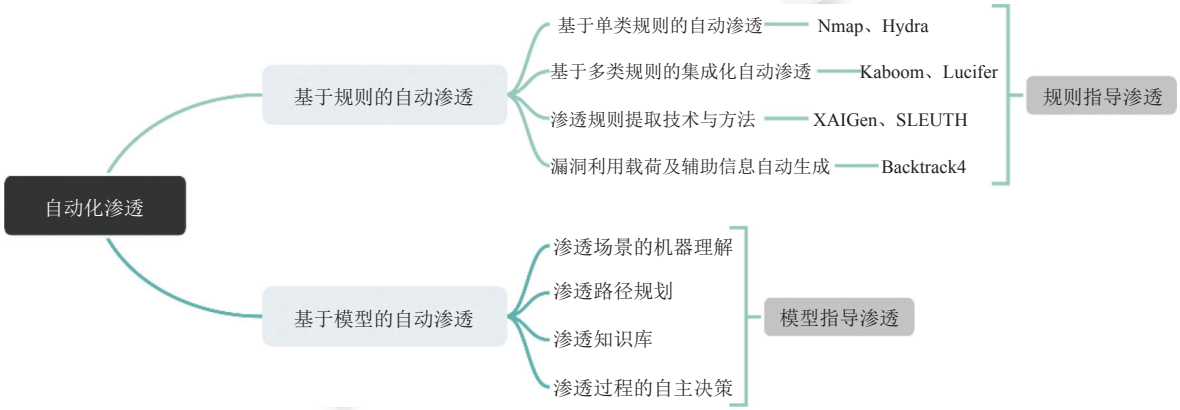


图 4 自动化渗透测试系统架构

2 基于规则的自动化渗透

渗透测试需要渗透测试人员进行大量重复流程. 为提高效率, 早期的自动化渗透系统是将固定的渗透步骤依据渗透经验组合成自动化渗透程序. 这类程序的特点是依据规则组合渗透步骤, 通过将主机发现、漏洞利用等渗透技术自动化, 形成可自动执行的单个渗透步骤, 然后根据预设规则, 依据渗透目标自主选择渗透策略和渗透方法, 从而实现渗透过程自动化, 在此过程中, 需重点解决渗透规则提取和漏洞利用载荷生成问题, 如图 5 所示.

图 5 基于规则的自动化渗透

根据规则类型, 基于规则的自动渗透系统分为单类规则和多类规则, 前者只执行某一战术, 功能简单且只针对特定任务, 目标是帮助安全管理人员对目标自动执行特定功能的渗透任务, 不足之处是一旦目标不匹配或者渗透参数配置错误时存在失败的风险. 后者则通过多个渗透战术规则实现多种渗透战术组合, 自动化程度更高, 但高度集成也带来渗透系统灵活性差, 渗透行为之间配合度较低的缺点, 且渗透参数需手动配置.

2.1 基于单类规则的自动渗透

基于单类规则的自动渗透技术将一系列相同动作的渗透步骤组合, 实现单个渗透战术的自动化执行, 如目标发现、初始访问和权限提升等, 常见的黑客工具 DCShadow[22]、Atbroker.exe[23]等均属于此类. 依据 ATT&CK 矩阵子技术可对基于单类规则的自动渗透系统进行清晰的类别划分, 如表 2 所示, 显然基于单类规则的自动渗透无法适应复杂场景下多个渗透技术组合利用的要求.

表 2 自动化渗透程序

工具名称	ATT&CK技术点 (技术编号)	渗透任务
powershell	修改认证过程 (T1556)	命令执行
Certutil.exe	混淆文件或信息 (T1027)	文件下载
DCShadow	非法域控制器 (T1207)	复制数据
Atbroker.exe	签名二进制代理 (T1218)	恶意程序启动
xattr(sh)	破坏信任控件 (T1553) 看门人绕过 (T1553.001)	扩展属性编辑
Office模板	模板注入 (T1221)	注入恶意木马
systemd-detect-virt(bash)	虚拟化/沙盒规避 (T1497) 系统检查 (T1497.001)	检测运行环境
Microsoft msxsl.exe	脚本处理 (T1220)	XSL处理

2.2 基于多类规则的集成化自动渗透

基于多类规则的集成化自动渗透由渗透模块、规则库、规则配置以及信息处理这 4 部分构成, 逻辑结构如图 6 所示, 系统依据收集得到的攻击辅助信息 (如拓扑、操作系统指纹等), 利用规则匹配从规则库中选择适当的渗透程序进行渗透步骤的组合. Kaboom[24]是 2020 年发布的典型的基于渗透脚本组合的自动化渗透系统, 它集成了 Hydra[25]、Nmap、Dirb[26]等工具, 可完成各阶段渗透测试中的不同任务, 如信息收集、漏洞评估及漏洞利用等. 知名工具 Metasploit[27]也属于此类, 它覆盖整个渗透流程功能, 自动完成木马下发、内网穿透等功能, 在渗透实战中应用非常广泛. Cobalt Strike[28]是一个著名的综合自动化渗透程序, 集成了端口转发、扫描、Windows 可执行程

序生成、Windows 动态链接库生成、Java 程序生成、Office 宏代码生成, 包括站点克隆获取浏览器的相关信息, 覆盖 ATT&CK 矩阵的多个技战术.

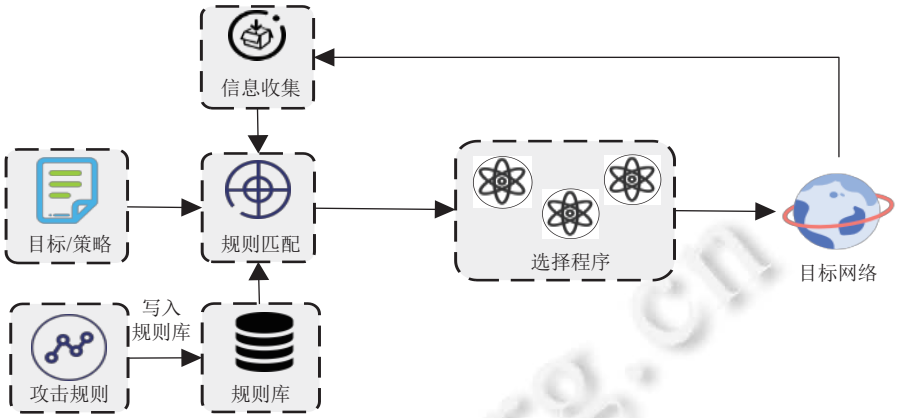


图 6 基于规则的自动化渗透系统

针对内网渗透场景, 结合 PTES 渗透测试执行标准和 ATT&CK 矩阵, 文献 [29] 设计了一种自动化内网突破方案, 该系统包含信息收集、漏洞探测、漏洞利用、权限提升、后渗透测试和痕迹清理这 6 个部分, 同类型系统还有 Lucifer[30]、AttackSurfaceMapper[31]、Nettacker[29]和 LazyRecon[32]等. 文献 [33] 则提出一种针对云服务程序的粗粒度安全评估自动化渗透测试系统, 此类针对特定应用的集成化自动渗透无需考虑网络环境多样性, 可有效提高可靠性和测试效率. 文献 [34] 将渗透经验转化为渗透规则, 结合渗透测试框架 Metasploit, 提出一个由 5 阶段组成的渗透测试模型. 文献 [35] 则做了更为广泛的集成, 设计并实现了一种基于 SNMP (simple network management protocol)[36]、多源漏洞库以及基于 NASL (nessus attack scripting language)[37]插件的自动化渗透测试系统, 可通过渗透规则描述选择更为适合的渗透插件.

基于多类规则的集成化自动渗透的共同特点是开发者使用规则描述渗透知识, 将重复的渗透步骤以脚本形式按预制顺序执行. 通过决策树表征的渗透规则是此类系统的核心, 每条决策树分支都存储了一个完整的渗透步骤, 自动渗透过程被描述为不断进行渗透决策的过程[38].

2.3 渗透规则提取技术与方法

渗透规则是攻防知识的高度抽象, 用来表征渗透过程中采取的行为动作、恶意软件、渗透工具等信息. 根据特征源, 渗透规则提取可分为基于渗透经验的渗透规则提取、基于威胁情报 (cyber threat intelligence, CTI) 的渗透规则提取、基于入侵检测系统 (intrusion-detection system, IDS) 的渗透规则提取, 三者之间的逻辑关系如图 7 所示.

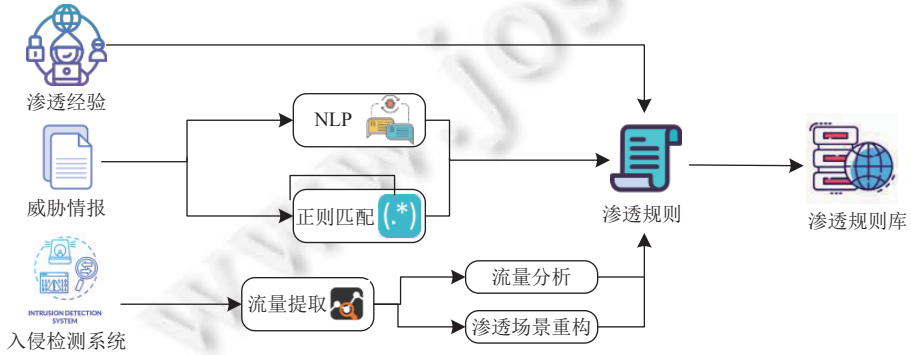


图 7 渗透规则提取框架

基于渗透经验的渗透规则提取技术, 依赖人工经验和目标场景分析来生成相应规则[39,40]. 其优点是对目标环境的适配性好, 规则使用的门槛较低而成功率更高; 其缺点也较为明显, 一是规则严重依赖专家经验; 二是规则库需频繁更新, 维护成本大. 文献 [41] 提出一种基于渗透测试本体的半自动知识提取方法, 将安全专家提出的概念 (流程) 抽象层和安全分析工具/方法/技术的抽象层相结合, 形成渗透知识, 从大量定期更新的异构数据中提取规则, 有效提高渗透测试自动化程度.

基于 CTI 提取渗透规则是现阶段研究热点, CTI 中包含有大量网络威胁细节, 诸如渗透步骤、被渗透系统的影响、威胁指标 (indicators of compromise, IOC) 及破坏指标等信息, 可利用自然语言处理技术 (natural language processing, NLP) 自动提取 CTI 报告中的渗透知识. 文献 [42] 使用文本识别和 NLP 技术收集高级持续性威胁 (natural language processing, APT) 的情报信息, 去除干扰信息后结合黑白名单和正则表达式提取 IOC, 使用 ATT&CK 矩阵技术提取战术、技术及过程, 并基于 BERT[43]威胁情报命名实体识别模型提取威胁实体. 文献 [44] 利用正则表达式抽取 CTI 中的 IOC, 使用命名实体识别模型 (named-entity recognition, NER)[45]识别 CTI 中的攻击实体, 并使用 NLP 模型提取实体内部的依赖构建攻击图; 文献 [46] 则提供一个基础库包含基于正则表达式以及 FLAIR[47] 和 SpaCy[48]模型的攻击实体提取方式, 试图解决 CTI 情报中渗透技术和攻击实体提取难题.

为提高 IDS 检测精准度, 有研究者利用机器学习和深度学习对未知攻击流量进行威胁特征自动提取[49,50], 以此扩充 IDS 对未知威胁的检测能力[51]. 开源项目 XAIGen[52]使用模型推断方法进行部分载荷的规则提取, 在去除干扰字符信息后对有效载荷进行聚类, 利用最长公共子序列算法和模型推理来提取漏洞有效载荷. 可实时重构渗透场景的 SLEUTH[53]系统, 则采用一种标签技术来分析渗透行为, 利用图存储依赖关系、先验知识以及重要文件评估事件, 解决高效事件存储和分析的问题. 文献 [54] 描述了安全知识图谱在自动化渗透行为提取上的应用, 使用知识图谱表示安全审计日志并基于 TransE[55]模型抽取行为实例, 采用层次聚类分析算法聚合行为实例. 文献 [56] 使用 NLP 中的深度学习算法, 训练威胁实体提取和关系提取模型, 最终构建出一个中文 CTI 实体关系数据集.

2.4 漏洞利用载荷及辅助信息的自动生成

漏洞利用载荷是影响渗透成功率的关键因素, 也是渗透测试中最为耗时的部分. 自动漏洞利用载荷生成可极大地提高漏洞利用效率. 文献 [57] 提出一种漏洞载荷自动化生成方案, 将载荷生成流程抽象为载荷构造、载荷变形, 针对输入的恶意函数调用序列构建分析模型, 自动生成载荷指令序列. 利用有限状态机实现广度优先搜索方法, 发现可用渗透技术并合并代码. 文献 [58] 使用符号执行技术, 及时灵活地发现 Web 应用程序中的各种漏洞类型, 文献 [59] 利用样本输入, 通过污点分析技术来探测漏洞. 模糊测试和符号执行技术正逐渐被应用于载荷自动生成研究领域[60–62].

以 CGC (cyber grand challenge)[63]为代表的漏洞利用自动化已不是一个新鲜的话题, AEG (automatic exploit generation)[64]目标是推动机器的自动攻防能力, 即自动分析测试程序并推理出漏洞位置, 并产生触发该漏洞的利用载荷. 主流分析方法分为静态分析、动态分析等. 静态分析包括对二进制进行句法分析、生成汇编指令的中间语言[65]以及构造控制流图[66], 动态分析则需核查程序在给定环境运行时的行为, 采用的技术包括二进制程序插装[67]、符号执行[68]以及模糊测试[69]等.

渗透执行过程中的状态信息可对后续渗透步骤提供正向反馈. 文献 [70] 基于贝叶斯推理实现渗透语义知识挖掘算法, 利用有向二分图模型表示渗透产生的参数信息, 并由此提取渗透语义知识. 文献 [71] 则利用 BackTrack4[72] 搭建测试平台, 格式化渗透产生的数据, 有效利用渗透过程中产生的渗透知识.

2.5 小 结

综上, 基于规则的自动化渗透系统将人工渗透测试经验转化为渗透规则, 并将一系列渗透动作组合成渗透程序. 这种系统的优点是渗透规则源自专家的实战经验, 可靠性较高, 对于适配的渗透场景决策速度快且渗透成功率高. 然而, 对于不适配的场景, 这种系统可能无法进行有效的渗透, 因为渗透规则的完备性和载荷数量是决定渗透效率和准确性的重要因素[73,74].

未来基于规则的自动化渗透测试研究应该聚焦于提高基于规则的自动化渗透系统的适应性和扩展性, 包括自动化生成和更新渗透规则、增强动态适应能力以及开发高级漏洞载荷生成方法, 以适应不断变化和复杂化的网络

攻防对抗环境随着人工智能技术的逐渐成熟, 利用 AI 模拟黑客攻击行为是当前的发展主流, 基于模型的自动渗透系统具有良好的自适应性, 可依据目标场景自动选择渗透方案与策略, 通过机器学习和深度学习算法, 可以不断学习和适应新的渗透技术和防御机制, 更新自身的知识库, 并生成新的渗透模式和策略, 从而展现出卓越的学习能力.

3 基于模型的自动化渗透

为提高自动化渗透系统的智能性, 进一步去除人工依赖, 研究者们开始尝试将 AI 技术与渗透技术相结合, 提出基于模型实现的自动化渗透系统, 通过关联目标场景与 AI 模型, 使渗透引擎可以感知目标场景变化, 并在此基础上规划渗透路径, 实现渗透过程的自主决策[75].

纵观以往研究成果, 基于模型的自动化渗透主要解决如图 8 所示的 4 个核心研究点: 渗透场景的机器理解、渗透路径规划、渗透知识库构建及渗透过程自主决策. 自动化渗透是这 4 个研究点的综合应用, 通过理解目标系统特征和漏洞信息, 系统能够确定最佳的渗透路径和策略. 构建渗透知识库提供渗透技术和工具资源支持. 在实际渗透测试过程中, 系统能够自主地做出渗透决策和行动规划. 这些组成部分相互依赖, 协同工作, 实现了智能、自适应和自主的自动化渗透测试, 提高了渗透测试的效率和准确性.

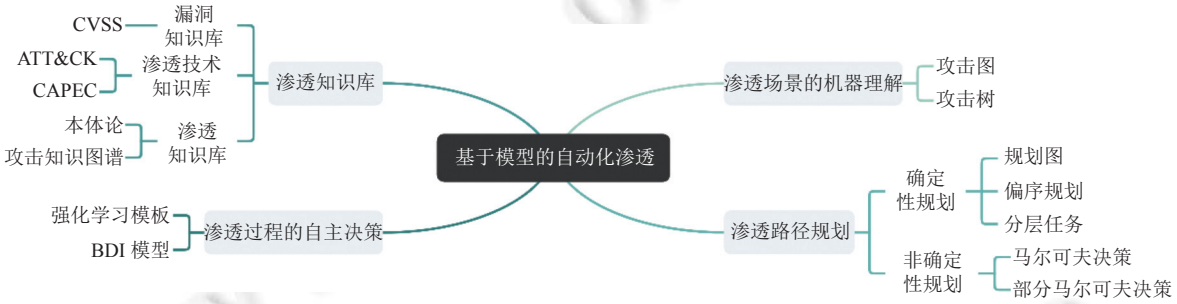


图 8 基于模型的自动化渗透框架

3.1 渗透场景的机器理解

渗透场景的机器理解是将主机 IP 地址、主机系统服务版本等物理网络信息转换为数字信息, 使用攻击树/图等方法描述渗透场景[76,77], 并依此制定渗透策略, 这些场景建模方法各有特点, 具体分析如下.

Schneier 于 1999 年率先采用树状结构表示系统渗透, 以便充分展示各渗透方法相互作用关系[78]. 攻击树是多层树状图, 可用于渗透推理, 每一个节点的渗透成立条件只和其相邻子节点有关, 若子节点渗透条件成立, 则父节点也具备渗透条件, 直到根节点渗透条件成立则表示渗透已成功. 2020 年 Hu 等人[79]使用多主机/多阶段分析将网络拓扑转化为攻击树, 辅助后续深度强化学习对渗透过程进行决策.

攻击图是一种网络脆弱性分析技术, 可自动分析目标网络各脆弱性之间的关系和由此产生的潜在威胁, 具体可分为状态攻击图和属性攻击图[80], 前者由于状态空间爆炸而较难实际应用. 属性攻击图常被用于分析漏洞之间的关系, NetKuang 是最早提出的完整攻击图生成方法[81], 目前被用来分析 UNIX 主机的网络配置脆弱性. 文献 [82] 将攻击图技术应用于计算机网络建模, 并指出渗透路径规划是攻击图技术的未来发展方向, 主要解决过多网络节点和复杂的网络连接使得攻击图生成速度变慢的问题. 2010 年为解决攻击图构造问题, Obes 等人[20]开发了 PDDL (planning domain definition language) 语言, 实现了基于 Metric-FF[83]和 SGPlan[84]的渗透策划器, 并集成到渗透测试工具中. 2012 年提出的最小化攻击图[85]以及 2022 年的自动化渗透系统[86]都采用攻击图技术实现渗透建模.

采用 Datalog[87]作为 bug 规范建模、配置描述、推理规则、操作系统权限和特权模型等信息的 MulVAL[88]是一个端到端的推理系统, 可在网络上进行多主机/多阶段漏洞分析. 通过为目标网络拓扑结构找到所有可达路径, 建立路径的状态转移矩阵, 并使用深度优先算法进行简化, 使得状态转移矩阵更加适合强化学习算法, 有效解决了

攻击图生成速度问题. Jajodia 等人提出的 TVA (topological vulnerability analysis) [89]是一个具有多项式级时间复杂度的攻击图生成工具, 常被用于自动化网络渗透分析, 其输出结果为由渗透步骤和渗透条件构成的状态攻击图, NetSPA (network security planning architecture)[90]是一种基于图论的攻击图生成工具, 使用防火墙规则和漏洞扫描结果构建网络模型, 并依次计算网络可达性和渗透路径. TVA 和 NetSPA 均需手动编写规则, 且状态空间爆炸问题并未得到妥善解决.

攻击图可以让模型理解网络场景, 但其计算开销大使得攻击图技术不具备实时规划能力, 无法在渗透测试过程中实时生成或更新攻击图, 也难以满足灰盒/黑盒渗透测试, 更无法处理渗透规划所涉及的各种不确定性, 渗透目标和系统状态经常发生变化, 而攻击图技术往往难以适应这种不确定性, 导致规划过程不够灵活和准确.

未来研究可以探索更高效的攻击图计算方法, 以实现更实时的规划能力. 同时, 结合其他技术, 如机器学习和数据驱动方法, 可以提高攻击图的自动化生成和更新能力. 此外, 引入风险评估和决策分析的方法, 可以更好地处理渗透规划中的不确定性. 综合利用这些技术和方法, 有望改进攻击图技术, 并使其更适用于实时渗透测试和处理各种不确定性情况.

3.2 渗透路径规划

渗透路径规划问题是依据网络实时状态发现渗透路径, 智能选择有效的漏洞利用载荷并配置载荷参数. 渗透路径规划[91]分为基于状态攻击图的渗透路径发现[92,93]、基于属性攻击图的渗透路径发现[94,95]. 具体是将渗透测试建模为攻击树[96]、攻击图[97], 或是将渗透行为进行编码, 将已知漏洞组合利用的挖掘问题, 转换为智能规划问题进行求解; 代表性的研究有 2016 年的 APTS 系统[98]使用 Petri 网[99]发现渗透路径, 2020 年的 ASAP 系统[100]使用强化学习算法解决经典规划问题, 并设计实现了一个安全分析和渗透测试框架.

上述工作均忽略了黑客攻击时信息不对称的特点[101], 且缺乏可扩展性, 进一步采用规划图等确定性规划算法以及马尔可夫模型、部分可观测马尔可夫模型等非确定性规划算法, 可有效实现渗透路径的快速发现, 但如何有效表征渗透知识和规划算法表达式之间的映射, 仍是一个亟待解决的热点问题.

确定性规划问题强调解决完全观测条件下的路径规划. 基于确定性规划算法的渗透路径路线发现, 首先将渗透知识转化为 PDDL 表示形式, 再利用规划算法确定渗透路径, 依据规划算法的不同, 可分为规划图技术[102]、偏序规划技术[103]和分层任务技术[104].

规划图技术利用交替的图扩展和解提取, 实现渗透路径规划. 图扩展利用前向搜索从当前可行的状态集合中查找可用的动作集合, 并生成下一步的可行状态集合, 继而判断状态集合是否满足要求, 更新集合状态集合和动作集合. 解提取则依据互斥条件, 对当前状态节点进行判断, 观察其是否包含目标状态集合. 若包含, 依据目标状态的前置条件更新目标状态集合并后向搜索直到获取可行路径, 否则对当前规划图进行图扩展操作. 将渗透计划编码到 PDDL 中, 利用智能规划算法发现渗透路径.

在使用规划描述语言的基础上, 基于偏序规划的路径发现利用目标和动作的结构使得路径搜索更加有效[105]. 偏序规划约束了必须存在先后关系的渗透动作, 任何满足偏序关系的规划路径都是目标路径. 偏序规划过程因需要多次遍历动作集合导致算法复杂度较高. 文献 [106] 将偏序规划算法合并到前向搜索框架中, 有效解决了时序数值规划问题.

基于分层任务网络的路径发现[107]重点在于基于任务与任务分解的方式实现渗透路径发现. 分层任务网络将渗透目标设置为原始任务, 非原子/原子任务表示不可/可以直接执行的任务状态. 核心思想是将原始任务分解为子任务, 若该子任务为原子任务, 则直接完成, 否则继续分解, 直到全部分解为可执行的原子任务进而推断出可行路径. 其优点在于对规划问题进行抽象分析, 忽略底层细节, 但原子任务的状态具体需要定制[108,109], 截至目前, 尚未见到相关技术的实际应用.

显然, 确定性规划算法的路径发现技术需利用搜索技术遍历路径空间发现可行解, 较为适合白盒渗透测试, 在黑盒状态下, 目标网络信息是部分可观测的, 如何在路径规划过程中考虑非确定条件, 采用马尔可夫 (Markov decision process, MDP) 和部分马尔可夫模型 (partially observable Markov decision processes, POMDP) 进行非确定

性条件下路径规划, 逐步成为本领域的研究热点.

马尔可夫决策过程的状态转移取决于当前状态和所采取的动作. 文献 [100] 将渗透测试流程形式化为 MDP 过程, 漏洞集合作为动作空间, 状态空间则由渗透动作和渗透结果组成, 奖励函数依赖于常状态转移量与损失值, 目标是最小化期望损失值. 在 MDP 过程的基础上, POMDP 过程增加了状态观测不确定性, 是环境状态部分可知动态不确定环境下序贯决策的理想模型, 其核心点在于模型不了解所处环境状态, 必须借助额外的感知模块, 或与其他模型交互方能获知状态. 基于 POMDP 建模技术, 文献 [101] 实现了一种自动导出多步渗透测试的规划方法. 近年来的自动化渗透研究将渗透测试问题建模为部分可观测马尔可夫决策, Zhou 等人[109]、Tran 等人[110]、Ghanem 等人[111,112]将渗透测试的过程建模为马尔可夫决策问题, 文献 [113] 提出使用 POMDP 建模渗透计划问题的方案能够依据得到的知识推理, 并且依据获取的网络信息赋能后续渗透动作, 而文献 [112] 提出一种依赖 POMDP 对单个目标的有效渗透方法, 通过将这些渗透组合, 形成对整个网络的渗透. 2020 年, Schwartz 等人[114]综合考虑防御者行为描述, 提出基于 POMDP 的自主渗透测试框架, 有效提升了系统的智能性.

非确定性规划考虑了渗透测试过程存在的不确定因素, 路径规划更加真实, 有效路径更多, 具有较好的研究前景, 但算法求解复杂, 提高求解效率是该类方法未来的研究重点.

3.3 渗透知识库

渗透知识库是自动化渗透的重要支撑, 是对渗透理论/技术深入理解基础上, 重点解决如何选取合适的表示方法对数量庞大渗透信息进行表征. 相关技术和方法有: 面向漏洞信息描述的 CVSS[115]、面向渗透技战术模型的 ATT&CK 及具有推理功能的本体论 (Ontology)[116]和安全知识图谱 (cyber security knowledge graph) 等.

CVSS 通用漏洞评分系统是依据漏洞利用条件、威胁程度等, 提出标准的描述方式建模漏洞的渗透成本、渗透收益等信息所构建的漏洞知识库. CVSS 计算基本维度 (base metric group), 时间维度 (temporal metric group) 和环境维度 (environmental metric group) 所得的分数, 组成 0–10 分的漏洞总体得分, 评估漏洞的严重程度. 文献 [117] 从攻防两个角度将网络攻防动作分解为渗透成本、渗透收益、防御成本和防御收益, 利用 CVSS 评分指标量化评估行为要素. 文献 [118] 基于 CVSS 提出面向工业互联网的漏洞分析框架. 文献 [119] 则提出利用 CVSS 结合图安全模型分析网络渗透, 获得潜在渗透路径.

面向渗透技战术的渗透知识表示模型 ATT&CK, 是从攻击者角度对网络攻击和入侵进行分类和描述的模型. 通过将已知的渗透行为转化为表征渗透技战术的结构化列表, ATT&CK 全面呈现了攻击者在网络攻击时采用的可能动作, 对于渗透模拟等具有非常重要的意义. ATT&CK 相比 Kill Chain[120]、CAPEC[121]等模型更加贴合实战. 可使用 ATT&CK 提供的技术作为动作集合指导自动化渗透, 依据 ATT&CK 的战术进行知识决策. 目前越来越多的研究者开始关注 ATT&CK, 例如原子红队[122]是通过脚本自动化, 对 ATT&CK 框架渗透手法的具体实现.

本体论原本是哲学领域概念, 研究客观事物存在的本质, 本体不仅允许通过形式化共享和重用领域知识, 还具有良好的概念层次结构和逻辑推理支持. 2004 年, Pinkstion 等人基于超过 4000 类计算机攻击事件生成了一个本体[123], 根据目标的系统组件、渗透手段、渗透结果和攻击者的位置进行分类, 实现渗透建模. 2007 年 Herzog 等人[124]提出基于 OWL (Web-ontology language) 语言对资产、威胁、漏洞、对策及其关系的模型, 能够推理各种主体之间的关系; 2009 年 Wang 等人[125]为了实现信息安全自动化, 基于国家漏洞数据库 (NVD) 构建了漏洞本体模型 (OVM), 并附加推理规则、知识表示和数据挖掘机制, OVM 能继承常见的漏洞及相关渗透展示, 为自动化渗透提供支撑; 2013 年, Gao 等人[126] 提出了基于渗透影响、渗透向量、渗透目标、漏洞和防御等维度的本体模型, 详细分析了各维度之间存在的关系, 使用 NVD 的漏洞信息填充本体, 相比于 OVM , 其描述的维度更侧重体现漏洞之间的关系. 2015 年, Stepanova 等人[41]利用渗透测试本体自动化从异构数据中提取知识, 利用提取的知识实现自动化渗透. 2020 年, Chu 等人[127]提出基于渗透分类的本体, 将攻击者、目标和攻击方法类构建为本体的顶级概念, 攻击者类包括一组攻击者实例, 将其作为自动化渗透的基础.

安全知识图谱是知识图谱在网络安全领域的具体应用, 对网络信息加工、处理、整合, 转化为结构化的安全领域知识库, 当前安全知识图谱构建大多结合已有的安全知识库进行构建, 如 CVE, CPE, CAPEC 和 ATT&CK 等.

STUCCO[128]通过采集安全系统中的数据, 提取领域概念和关系, 并将这些信息集成到网络安全知识图中以加速决策制定. 文献 [129] 使用知识图技术构建了基于真实渗透场景的 APT 知识图模型, 为威胁情报与信息提取信息提供了解决方案, 应用知识图模型提出 APT 防御方法. 安全知识图谱包含的渗透信息如何指导自动化渗透, 将是另一个研究热点.

3.4 渗透过程的自主决策

基于规则的系统不会依据环境选择渗透策略, 而基于模型的渗透则要求在渗透的过程中学会依照环境选择不同策略. 渗透过程自主决策通常使用的模型是强化学习模型和 BDI (belief-desire-intention) 模型[130]. 基于模型的自动化渗透框架如图 9 所示, 系统接收网络资产信息后, 建立攻击图, 作为智能体的环境信息, 并利用强化学习输出渗透路径, 并利用渗透工具渗透真实的网络.

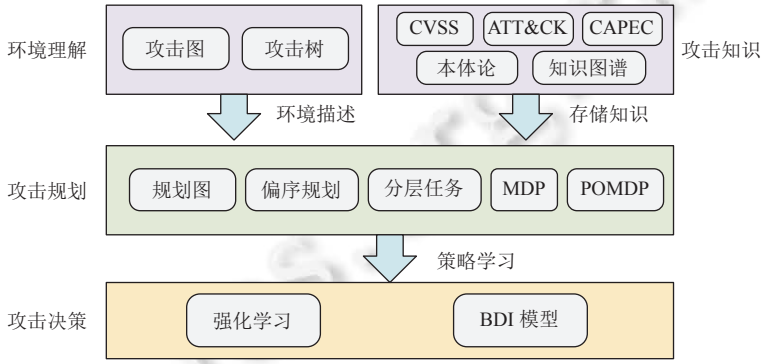


图 9 基于模型的自动化渗透框架

强化学习[131]强调如何基于环境而行动, 以最大化预期利益, 是除了监督学习和非监督学习之外的第 3 种机器学习方法. 强化学习基本流程为: 智能体 (Agent) 通过观察环境的状态做出行动, 该行动会作用于环境, 改变环境状态, 并且产生相关联奖励, 智能体通过观察新的状态和奖励来进行下一步动作. 在上述循环过程中, Agent 会不断得到奖励, 从而不断进化, 最终能以利益最大的目标实施行动. 强化学习对于已知网络环境的渗透路径发现非常有效, 强化学习模型对于解决模型未知的 MDP 具有非常好的效果, 非常适合网络安全不断变化的性质. 强化学习在自动化渗透中的应用带来了自主决策、学习适应性和探索未知的能力, 使系统能够自动选择最佳渗透行为, 不断调整和优化策略以适应不同的渗透测试环境, 同时主动探索新的渗透路径和漏洞. 这种自动化能力提高了渗透测试的效率和准确性, 帮助发现并解决系统中的安全弱点. 强化学习技术衍生了许多研究成果和商业解决方案, 同时也有较多的开源实现工具[132].

网络渗透的动作空间是巨大且离散的, 强化学习能否成功应用于自动化渗透, 首先需要回答如何在离散的动作空间中选取相应渗透动作. 将强化学习用于自动化渗透最早是由 Greenwald 等人于 2009 年提出[133]. 作为改进, 文献 [134] 提出利用模型观察主机或子网制定渗透策略访问隐藏资产并渗透目标. 2015 年, Dulac-Arnold[135]利用 RL (reinforcement learning) 的 actor-critic 框架解决了强化学习动作空间过大的问题. 相关研究还包括: 通过共享决策模块等方法解决多动作维度在强化学习中的应用[136]、采用非常规 DRL 架构来解决渗透测试动作空间设置[110]、基于深度 Q 网络 (DQN) 并使用分层代理强化学习方法的显式动作分解方案[137]、结合深度强化学习和后渗透的框架的 PowerShell Empire[138]等.

使用 A2C[139]、Q-learning[140] 和 SARSA[141]这 3 种强化学习模型, 能够获得域控服务器的控制权限, 基于强化学习的自动化渗透研究包括基于 DQN 的自动渗透测试框架[75]、自动化攻防系统[114]、横向移动方法[142]、自动后渗透方案[143]及 DeepExploit[144]、AutoPentest-DRL[145]、OpenVAS[146]等, 这些研究都是基于强化学习实现的自动化渗透技术.

BDI 模型是对智能代理程序进行编程而开发的软件模型, 使用知识库和规则描述实现渗透信息推理. 表面上

以实现代理程序的信念、愿望和意图为特征, 实际上使用这些概念来解决代理程序编程中的特定问题. 因此, BDI 模型能够平衡在审议计划和执行这些计划上花费的时间. BDI 架构在自动化渗透测试的应用提供了智能决策、灵活适应以及行动规划的能力, 通过代理对目标系统的信念、欲望和意图的推理和决策, 自动化渗透测试系统能够自主地制定渗透策略和计划, 根据环境的变化进行适应性调整, 并执行具体的渗透行动. 这种架构使得渗透测试系统能够以智能、灵活和自主的方式进行自动化渗透, 解决了决策制定、适应性和行动规划等关键问题, 提高了渗透测试的效率和智能性.

基于 BDI 模型的自动化渗透系统将网络环境信息视为信念 (belief), 将渗透目标希望得到的权限状态视为愿望 (desire), 渗透行为则被视为意图 (intention). 使用强化学习能够实现较为理想的自动化渗透测试系统, 但如果没有先验知识, 基于强化学习的渗透更像是暴力测试, 使用 BDI-agent 能够依据指定的规则信息和知识库依据目标推理渗透方式. Qian 等人[147]提出的渗透框架基于 BDI-agent, 使用本体论描述渗透知识优化不确定环境中的规划问题, 并使用 SWRL[148]规则语言创建知识库并推理渗透知识, 利用 BDI 模型进行学习和推理从而实现自动化渗透测试, 类似的文献 [127] 提出利用 Protégé[149]创建渗透本体描述渗透知识, 同样使用 SWRL 语言创建知识库和推理条件. 针对渗透规划过程中的不确定性. 文献 [41] 提出一个可以捕捉结果中的不确定性的渗透规则模型, 利用每个渗透行为的成功率进行建模, 实现一个有效的规划算法来解决渗透路径的规划问题, 这个模型能够在几百个主机的场景下生成渗透计划.

BDI 技术支持灵活的层次化决策制定, 适用于需要逻辑推理和规则驱动的任务, 不需要进行大量的计算选择, 其运算速度更快, 但是这些推理规则也局限了渗透的行为和策略, 无法产生针对未知场景选择意图. 相比之下, 强化学习模型具备学习能力, 能够通过与环境的交互自主学习和改进决策策略, 并在未知环境中适应性强, 尽管强化学习方法也存在数据效率低和需要定义奖励信号的问题, 强化模型技术在智能渗透中具有更大的应用潜力.

3.5 小 结

基于模型的自动化渗透利用攻击图或攻击树等技术实现渗透场景的机器理解, 基于渗透知识库和规划算法选择渗透路径, 利用模型算法学习渗透策略. 基于模型的自动化渗透技术优点是能够使用智能模型实现对人决策行为的模拟, 自动选择渗透路径, 具有高智能性的优点. 针对使用模型的自动化渗透系统, 从使用的模型算法、渗透建模的规则、实现的渗透步骤、针对的渗透目标、渗透知识的表示、渗透场景感知方法等方面对现有的学术成果分析如表 3 所示.

表 3 基于模型的自动化渗透系统小结

文献	渗透场景的机器理解	渗透路径规划方式	渗透过程的自主决策	特点			
				多漏洞	单漏洞	策略规划	后渗透
[150]	格式化主机信息	马尔可夫决策	强化学习	√	—	—	—
[111]	格式化主机信息	部分可观察马尔可夫决策	强化学习模型	√	—	—	—
[133]	格式化主机信息	部分可观察马尔可夫决策	无	—	√	—	—
[79]	攻击图	无	强化学习	√	—	—	—
[143]	无	无	深度强化学习	—	—	—	√
[98]	Petri网	经典规划	渗透测试环境描述语言(PTEDL)	—	√	—	—
[151]	无	无	威胁模型	—	—	√	—
[109]	网络信息增益	马尔可夫决策	深度强化学习	√	—	—	—
[147]	本体论	SWRL	BDI-Agent	√	—	—	—
[127]	Protégé本体论	SWRL	BDI-Agent	√	—	—	—
[20]	攻击图	经典规划	PDDL	√	—	—	—
[114]	格式化主机信息	部分可观察马尔可夫决策	信息衰减因子	√	—	—	—
[137]	格式化主机信息	马尔可夫决策	深度强化学习	√	—	—	—
[152]	ATT&CK	部分可观察马尔可夫决策	LAVA	√	—	—	—
[153]	格式化主机信息	无	BDI-Agent	√	—	—	—

表 3 基于模型的自动化渗透系统小结 (续)

文献	渗透场景的机器理解	渗透路径规划方式	渗透过程的自主决策	特点			
				多漏洞	单漏洞	策略规划	后渗透
[154]	攻击图	经典规划	强化学习	—	√	—	—
[155]	无	马尔科夫决策	CLAP深度强化学习	√	—	—	—
[156]	敏感数据模型	无	无	—	√	—	—
[157]	深度强化学习	经典规划	无	√	—	—	—
[158]	威胁模型	经典规划	无	√	—	—	—

相比于基于规则的自动渗透, 基于模型的自动渗透缺点表现在大型复杂网络中的渗透决策能力差, 渗透行为过多时就会造成渗透效率低的问题, 但智能模型在渗透的应用能够实现渗透行为的自动学习和补充, 在面对未知环境时能够做出决定, 从而实现系统面对不同场景时都会做出合适的渗透策略. 机器学习技术已在自动化渗透领域凸显了其可行性与性能优越性, 基于机器学习的自动化渗透也成为了网络安全领域的一个研究方向.

4 总结与展望

本文详细介绍了自动化渗透的实现方案. 自动化渗透的研究起源于手动渗透, 通过特殊请求实现对目标的渗透测试; 在传统手动渗透的基础上, 基于简单规则形成渗透系统, 减少重复工作; 为加强系统智能性, 研究者们引入 AI 技术适应不同的渗透场景.

自动化渗透是网络安全研究方向的一个非常重要的方向. 一方面自动化渗透能够简化网络渗透, 降低渗透人员的技术门槛, 增加网络渗透的方式种类, 提升渗透的速度, 减轻渗透人员的工作量, 具有必要性; 另一方面, 随着人工智能技术的发展, 利用人工智能渗透目标网络已经具备理论条件, 具有可行性. 在 ChatGPT 这样的大型模型的背景下, AI 的能力已经证明可以在各种领域中发挥作用. 在未来的网络攻防对抗领域, 基于模型的自动化渗透测试将成为热点研究领域.

依据上文提出基于模型的自动化渗透需要解决的渗透场景的机器理解、渗透路径规划、渗透知识库以及渗透过程的自主决策这 4 个需要解决的问题, 研究者可以探索更高级的人工智能技术, 以提升模型的智能化水平. 研究者将进一步探索如何将强化学习应用于智能渗透中, 通过使渗透测试系统能够自主学习和优化其行为, 系统可以在动态和复杂的环境中不断提高其攻御策略, 以适应新的渗透技术和安全防御措施, 提升渗透过程的自主决策能力. 研究者可以聚焦在自适应渗透路径规划算法, 根据目标系统的特征和网络环境的变化, 智能地选择和调整渗透路径, 以最大程度地提高渗透成功率并减少被检测的风险.

在上述研究中, 大都是自动渗透无防护的目标网络, 不需要自动化渗透系统具有对抗意识, 但在实际的攻防渗透中, 则必须考虑自动渗透对抗的问题. 研究人员可以开发对抗性攻防技术, 使自动化渗透测试系统能够克服目标网络的防御机制. 通过引入对抗性学习方法, 系统能够具备对抗意识, 自主学习和调整渗透策略. 这些研究方向将帮助提高自动渗透系统在实际场景中的逼真性和实用性, 从而更好地应对对抗性环境的挑战. 研究者可以研究如何实现自动化修复和反击机制. 这意味着智能渗透测试系统不仅能够检测问题, 还能够自动修复或响应渗透, 以提高系统的弹性和恢复能力.

基于文献和团队的研究成果, 本文提出以下研究方向以供参考.

4.1 基于 RPC 的渗透框架

为实现模拟真实的网络渗透, 需要使用多种网络渗透工具, 但安全工具种类繁多且同质化严重, 实际渗透测试中存在大量无序性、重复性的工作, 极大地限制了测试效率. 原子模块协同的渗透测试框架, 主要思想是分析渗透测试过程中各主要阶段需求来构造无系统依赖性的原子渗透模块和实现信息传递标准化. 网络渗透中通常包含多个阶段, 通过构造无系统依赖性的原子渗透技术, 依据各阶段特点和阶段间的逻辑关系, 明确划分原子模块功能, 利用虚拟化技术部署服务, 进而以并联、串联、异构、复合等方案自由组合和拓展, 可有效提高渗透行为的复杂

性和多样性. 模块间协同工作需要制定与应用统一的信息分类分级、记录格式及其转换、编码等技术标准, 以实现信息传递标准化是信息共享和系统兼容. 利用 RPC (remote procedure call) 远程调用虚拟化的渗透程序, 对目标网络实现分布式渗透, 如图 10 所示.

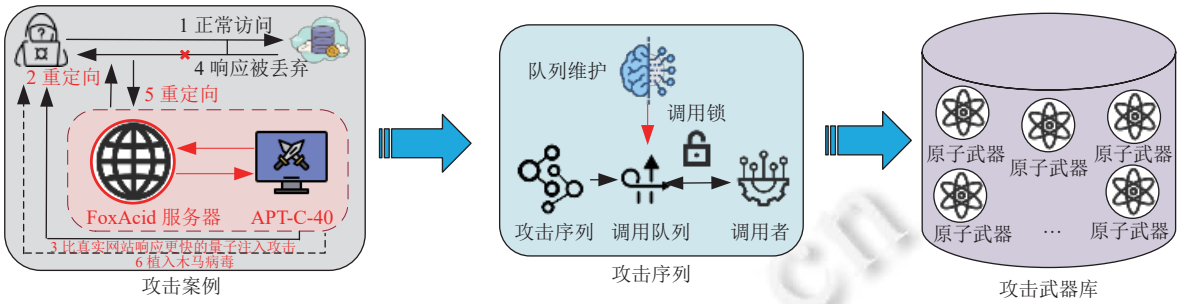


图 10 基于 RPC 的网络渗透框架

4.2 动作空间构建与渗透动作优选

渗透动作是在渗透目标时使用的行为, 包括访问目标系统、执行命令、隐藏渗透痕迹等. 类比迷宫探索类的问题, 渗透也是一类探索目标并依据目标状态选择渗透行为, 需要注意的是迷宫探索类问题其动作行为是局限的, 迷宫内的行动空间是可以被穷尽的, 可简单划分为前后左右这 4 个方向, 而网络空间的渗透行为是无限的, 所使用的渗透方式无法被简单穷尽. 因此, 合理构建渗透的动作空间是实现自动渗透的一个重要问题. 首先需要解决的是所构建动作空间的渗透行为来源, 可能的渗透工作行为来源包括渗透流量、渗透日志等, 其次是渗透动作的粒度选择. 过小的渗透粒度使得动作空间的总量过大, 计算需要的资源更多耗时更久; 过大的渗透动作粒度不利于适配复杂场景, 合适的渗透粒度可以适配部分相同和相似的渗透场景, 而不必要求完全相同的渗透目标环境. 在得到可用的渗透动作后, 最后是渗透动作优选问题. 需要明确的是在攻防对抗的过程中, 越多的渗透动作越容易被防守方发现, 因此渗透人员需要在已有可行的渗透行为集合内选择能够达到相同渗透目的的最小集合并且将其最小化, 能够减少网络上的渗透流量痕迹, 隐藏渗透行为.

4.3 漏洞组合利用

对渗透测试来说, 多个漏洞组合利用的威胁程度远远超过单个漏洞, 常见方法是组合利用一些低威胁的漏洞, 构造高危渗透行为, 例如使用某些低微的逻辑漏洞使得防护设备发出无效警告, 进而引导防护人员关闭防护告警, 从而实现高威胁渗透而不被防护设备告警. 因此, 可以看出利用多个漏洞组合造成的威胁超单个漏洞, 自动渗透实现高威胁渗透需要解决多个漏洞组合构造渗透链的问题. 在已经确定渗透行为集合和漏洞集合的条件下, 研究漏洞之间的关联关系, 从而明确漏洞之间的组合利用, 结合机器学习算法, 从而推导出在未知的组合利用方式, 是漏洞关联研究的研究目标.

References:

[1] The state of security 2023 is resilient. 2023. https://www.splunk.com/en_us/campaigns/state-of-security.html
[2] Cybersecurity threatscape: q1 2021. 2021. https://www.ptsecurity.com/ww-en/analytics/cybersecurity-threatscape-2021-q1/
[3] NETSCOUT DDoS threat intelligence report—Latest cyber threat intelligence report. 2023. https://www.netscout.com/threatreport
[4] Wang X, Gui CN. Observation on the situation of network security vulnerabilities in the first half of 2022. China Information Security, 2022(9): 85–87 (in Chinese). [doi: 10.3969/j.issn.1674-7844.2022.09.029]
[5] Thornberry WMM. National defense authorization act for fiscal year 2021. 2021. https://www.congress.gov/116/plaws/publ283/PLAW-116publ283.pdf
[6] Biden JR. Interim national security strategic guidance. 2021. https://www.whitehouse.gov/wp-content/uploads/2021/03/NSC-1v2.pdf
[7] Biden JR. Executive order on improving the nation's cybersecurity. 2021. https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/
[8] Teoh CS, Mahmood AK. National cyber security strategies for digital economy. In: Proc. of the 2017 Int'l Conf. on Research and

Innovation in Information Systems (ICRIIS). Langkawi: IEEE, 2017. 1–6. [doi: 10.1109/ICRIIS.2017.8002519]

[9] Cybersecurity law of the People's Republic of China. 2016 (in Chinese). http://www.cac.gov.cn/2016-11/07/c_1119867116.htm

[10] Regulations on the security protection of key information infrastructure (Order No. 745 of the state council of the People's Republic of China). 2021 (in Chinese). http://www.gov.cn/zhengce/content/2021-08/17/content_5631671.htm

[11] Computer security worries military experts. 1983. https://www.nytimes.com/1983/09/25/us/computer-security-worries-military-experts.html

[12] Stefinko Y, Piskozub A, Banakh R. Manual and automated penetration testing. Benefits and drawbacks. Modern tendency. In: Proc. of the 13th Int'l Conf. on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET). Lviv: IEEE, 2016. 488–491. [doi: 10.1109/TCSET.2016.7452095]

[13] Nmap: The network mapper. 2023. https://nmap.org/

[14] Download nessus vulnerability assessment. 2023. https://www.tenable.com/products/nessus

[15] Herzog P. Open-source security testing methodology manual. Institute for Security and Open Methodologies (ISECOM), 2003.

[16] OISS Group. Information Systems Security Assessment Framework. Open Information Systems Security Group, 2006.

[17] The PTES Team. The penetration testing execution standard documentation. 2022. https://buildmedia.readthedocs.org/media/pdf/pentest-standard/latest/pentest-standard.pdf

[18] Strom BE, Applebaum A, Miller DP, Nickels KC, Pennington AG, Thomas CB. MITRE ATT&CK: Design and philosophy. Technical Report, 10AOH08A-JC, The MITRE Corporation, 2018.

[19] Boddy M, Gohde J, Haigh T, Harp S. Course of action generation for cyber security using classical planning. In: Proc. of the 15th Int'l Conf. on Automated Planning and Scheduling. Monterey: AAAI Press, 2005. 12–21.

[20] Obes JL, Sarraute C, Richarte G. Attack planning in the real world. arXiv:1306.4044, 2013.

[21] Sun YB, Tian ZH, Li MH, Zhu CS, Guizani N. Automated attack and defense framework toward 5G security. IEEE Network, 2020, 34(5): 247–253. [doi: 10.1109/MNET.011.1900635]

[22] DCShadow attack. 2023. https://www.dcshadow.com/

[23] What is Atbroker.exe used for? 2023. https://file.info/windows/atbroker_exe.html

[24] leviathan36/Kaboom. 2023. https://github.com/Leviathan36/kaboom

[25] Hydra. 2023. https://www.kali.org/tools/hydra/

[26] Dirb. 2023. https://www.kali.org/tools/dirb/

[27] Metasploit. 2023. https://www.metasploit.com/

[28] Cobalt strike. 2023. https://www.cobaltstrike.com/

[29] OWASP/Nettacker. 2023. https://github.com/OWASP/Nettacker

[30] skiller9090/Lucifer. 2023. https://github.com/Skiller9090/Lucifer

[31] AttackSurfaceMapper. 2023. https://0x1.gitlab.io/reconnaissance-tools/AttackSurfaceMapper/

[32] nahamsec/LazyRecon. 2023. https://github.com/nahamsec/lazyrecon

[33] Casola V, De Benedictis A, Rak M, Villano U. Towards automated penetration testing for cloud applications. In: Proc. of the 27th IEEE Int'l Conf. on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE). Paris: IEEE, 2018. 24–29. [doi: 10.1109/WETICE.2018.00012]

[34] Yan JL. A research on automatic penetration testing based on Metasploit framework. Netinfo Security, 2013(2): 53–56 (in Chinese with English abstract). [doi: 10.3969/j.issn.1671-1122.2013.02.014]

[35] Bu YJ, Wang H, Hu JP, Zhang Q. Design and research of an automated security penetration testing system. Network Security, 2020(7): 32–34 (in Chinese). [doi: 10.3969/j.issn.1009-6833.2020.07.022]

[36] Harrington D, Presuhn R, Wijnen B. An architecture for describing simple network management protocol (SNMP) management frameworks. 2002. https://www.rfc-editor.org/info/rfc3411

[37] Rogers R. Nessus Network Auditing. 2nd ed., Burlington: Syngress, 2008.

[38] Zhao JM, Shang WL, Wan M, Zeng P. Penetration testing automation assessment method based on rule tree. In: Proc. of the 2015 IEEE Int'l Conf. on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER). Shenyang: IEEE, 2015. 1829–1833. [doi: 10.1109/CYBER.2015.7288225]

[39] Ge X, Yue MN, Jin JT. Automatic penetration testing framework based on campus network. Journal of Shenzhen University (Science & Engineering), 2020, 37(S1): 68–72 (in Chinese with English abstract). [doi: 10.3724/SP.J.1249.2020.99068]

[40] Chen SQ. The military communication network security research based on automated penetration testing [MS. Thesis]. Lanzhou: Lanzhou University, 2012 (in Chinese with English abstract).

[41] Stepanova T, Pechenkin A, Lavrova D. Ontology-based big data approach to automated penetration testing of large-scale heterogeneous systems. In: Proc. of the 8th Int'l Conf. on Security of Information and Networks. Sochi: ACM, 2015. 142–149. [doi: 10.1145/2799979.2799995]

[42] Zhou YH, Tang Y, Yi M, Xi CY, Lu H. CTI view: APT threat intelligence analysis system. Security and Communication Networks, 2022, 2022: 9875199. [doi: 10.1155/2022/9875199]

[43] Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805, 2019.

[44] Li Z, Zeng J, Chen Y, *et al*. AttacKG: Constructing technique knowledge graph from cyber threat intelligence reports. In: Proc. of the 2022 European Symposium on Research in Computer Security. Cham: Springer Int'l Publishing, 2022. 589–609.

[45] Mohit B. Named entity recognition. In: Zitouni I, ed. Natural Language Processing of Semitic Languages. Berlin, Heidelberg: Springer, 2014. 221–245. [doi: 10.1007/978-3-642-45358-8_7]

[46] Alam MT, Bhusal D, Park Y, Rastogi N. CyNER: A Python library for cybersecurity named entity recognition. arXiv:2204.05754, 2022.

[47] Akbik A, Bergmann T, Blythe D, Rasul K, Schweter S, Vollgraf R. FLAIR: An easy-to-use framework for state-of-the-art NLP. In: Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics (Demonstrations). Minneapolis: Association for Computational Linguistics, 2019. 54–59. [doi: 10.18653/v1/N19-4010]

[48] SpaCy. 2023. https://spacy.io/

[49] Wu SX, Banzhaf W. The use of computational intelligence in intrusion detection systems: A review. Applied Soft Computing, 2010, 10(1): 1–35. [doi: 10.1016/j.asoc.2009.06.019]

[50] Tabassum A, Erbad A, Lebda W, Mohamed A, Guizani M. FEDGAN-IDS: Privacy-preserving ids using GAN and federated learning. Computer Communications, 2022, 192: 299–310. [doi: 10.1016/j.comcom.2022.06.015]

[51] Wang CH, Chen J, Su H, He K, Du RY. Mobile advertising loophole attack technology based on host APP's permissions. Ruan Jian Xue Bao/Journal of Software, 2018, 29(5): 1392–1409 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5494.htm [doi: 10.13328/j.cnki.jos.005494]

[52] Yu XN, Guo WK, Liu YZ, *et al*. An automatic features extraction model of IDS for IoT. In: Proc. of the 2022 Int'l Conf. on Computer Engineering and Networks. Singapore: Springer, 2022. 1260–1268.

[53] Hossain MN, Milajerdi SM, Wang J, *et al*. SLEUTH: Real-time attack scenario reconstruction from COTS audit data. In: Proc. of the 26th USENIX Security Symp. 2017. 487–504.

[54] Leichtnam L, Totel E, Prigent N, *et al*. Sec2graph: Network attack detection based on novelty detection on graph structured data. In: Proc. of the 2020 Detection of Intrusions and Malware, and Vulnerability Assessment. Springer, 2020. 238–258.

[55] Bordes A, Usunier N, Garcia-Durán A, Weston J, Yakhnenko O. Translating embeddings for modeling multi-relational data. In: Proc. of the 26th Int'l Conf. on Neural Information Processing Systems. Lake Tahoe: Curran Associates Inc., 2013. 2787–2795.

[56] Zhou YH, Ren YT, Yi M, Xiao YJ, Tan ZY, Moustafa N, Tian ZH. CDTier: A Chinese dataset of threat intelligence entity relationships. IEEE Trans. on Sustainable Computing, 2023, 8: 1–13. [doi: 10.1109/TSUSC.2023.3240411]

[57] Cheng Y. Research and implementation of automatic payload generation system [MS. Thesis]. Xi'an: Xidian University, 2017 (in Chinese with English abstract).

[58] Shih HY, Lu HL, Yeh CC, Hsiao HC, Huang SK. A generic Web application testing and attack data generation method. In: Peng SL, Wang SJ, Balas VE, Zhao M, eds. Security with Intelligent Computing and Big-data Services. Springer, 2018. 232–247. [doi: 10.1007/978-3-319-76451-1_22]

[59] Kieyzun A, Guo PJ, Jayaraman K, Ernst MD. Automatic creation of SQL injection and cross-site scripting attacks. In: Proc. of the 31st IEEE Int'l Conf. on Software Engineering. Vancouver: IEEE, 2009. 199–209. [doi: 10.1109/ICSE.2009.5070521]

[60] Lee Y, Min C, Lee B. ExpRace: Exploiting kernel races through raising interrupts. In: Proc. of the 30th USENIX Security Symp. 2021. 2363–2380.

[61] Wang Y, Zhang C, Zhao ZX, Zhang BL, Gong XR, Zou W. MAZE: Towards automated heap Feng Shui. In: Proc. of the 30th USENIX Security Symp. 2021. 1647–1664.

[62] Park S, Kim D, Jana S, Son S. FUGIO: Automatic exploit generation for PHP object injection vulnerabilities. In: Proc. of the 31st USENIX Security Symp. 2022. 197–214.

[63] Song J, Alves-Foss J. The DARPA cyber grand challenge: A competitor's perspective. IEEE Security & Privacy, 2015, 13(6): 72–76. [doi: 10.1109/MSP.2015.132]

[64] Avgerinos T, Cha SK, Rebert A, Schwartz EJ, Woo M, Brumley D. Automatic exploit generation. Communications of the ACM, 2014, 57(2): 74–84. [doi: 10.1145/2560217.2560219]

[65] Brumley D, Jager I, Avgerinos T, Schwartz EJ. BAP: A binary analysis platform. In: Proc. of the 23rd Int'l Conf. on Computer Aided Verification. Snowbird: Springer, 2011. 463–469. [doi: 10.1007/978-3-642-22110-1_37]

[66] Bruschi D, Martignoni L, Monga M. Detecting self-mutating malware using control-flow graph matching. In: Proc. of the 3rd Int'l Conf. on Detection of Intrusions and Malware & Vulnerability Assessment. Berlin: Springer, 2006. 129–143. [doi: 10.1007/11790754_8]

[67] Nethercote N, Seward J. Valgrind: A framework for heavyweight dynamic binary instrumentation. ACM SIGPLAN Notices, 2007, 42(6): 89–100. [doi: 10.1145/1273442.1250746]

[68] Baldoni R, Coppa E, D'elia DC, Demetrescu C, Finocchi I. A survey of symbolic execution techniques. ACM Computing Surveys, 2018, 51(3): 50. [doi: 10.1145/3182657]

[69] Godefroid P, Levin MY, Molnar DA. Automated whitebox fuzz testing. In: Proc. of the 2008 Network and Distributed Systems Security. 2008, 8: 151–166.

[70] Zang YC, Hu TR, Zhou TY, Deng WJ. An automated penetration semantic knowledge mining algorithm based on Bayesian inference. Computers, Materials & Continua, 2021, 66(3): 2573–2585. [doi: 10.32604/cmc.2021.012220]

[71] Wen GX, Zhang YC, Zhang YQ. Automatic penetration test framework based on unified data format mechanism. Journal of University of Chinese Academy of Sciences, 2011, 28(5): 676–683 (in Chinese with English abstract).

[72] BackTrack Linux. 2023. https://www.backtrack-linux.org/

[73] Gao HJ, Li SM. Penetration test system based on Automation. Intelligent Computer and Applications, 2020, 10(4): 162–164 (in Chinese with English abstract). [doi: 10.3969/j.issn.2095-2163.2020.04.040]

[74] Xing B, Gao L, Sun Q, Yang W. Design and implementation of automated penetration testing system. Application Research of Computers, 2010, 27(4): 1384–1387 (in Chinese with English abstract). [doi: 10.3969/j.issn.1001-3695.2010.04.048]

[75] McKinnel DR, Dargahi T, Dehghantanha A, Choo KKR. A systematic literature review and meta-analysis on artificial intelligence in penetration testing and vulnerability assessment. Computers & Electrical Engineering, 2019, 75: 175–188. [doi: 10.1016/j.compeleceng.2019.02.022]

[76] Lu JJ, Huang LS, Wu SF. An attack tree approach for network intrusion modeling. Computer Engineering and Applications, 2003, 39(27): 160–163 (in Chinese with English abstract). [doi: 10.3321/j.issn:1002-8331.2003.27.051]

[77] Tidwell T, Larson R, Fitch K, Hale J. Modeling Internet attacks. In: Proc. of the 2001 IEEE Workshop on Information Assurance and Security. New York: IEEE, 2001. 54–59.

[78] Schneier B. Attack trees. Dr. Dobb's Journal, 1999, 24(12): 21–29.

[79] Hu ZG, Beuran R, Tan YS. Automated penetration testing using deep reinforcement learning. In: Proc. of the 2020 IEEE European Symp. on Security and Privacy Workshops (EuroS&PW). Genoa: IEEE, 2020. 2–10. [doi: 10.1109/EuroSPW51379.2020.00010]

[80] Chen F, Zhang Y, Su JS, Han WB. Two formal analyses of attack graphs. Ruan Jian Xue Bao/Journal of Software, 2010, 21(4): 838–848 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3584.htm [doi: 10.3724/SP.J.1001.2010.03584]

[81] Baldwin RW. Rule based analysis of computer security. 1988. https://hdl.handle.net/1721.1/149665

[82] Wang GY, Wang HM, Chen ZJ, Xian M. Research on computer network attack modeling based on attack graph. Journal of National University of Defense Technology, 2009, 31(4): 74–80 (in Chinese with English abstract). [doi: 10.3969/j.issn.1001-2486.2009.04.015]

[83] Metric-FF homepage. 2023. https://fai.cs.uni-saarland.de/hoffmann/metric-ff.html

[84] Hsu CW, Wah BW, Huang RY, Chen YX. New features in SGPlan for handling preferences and constraints in PDDL3.0. In: Proc. of the 5th Int'l Planning Competition. 2006. 39–42.

[85] Xie DQ, Li GC. Automated penetration test model base on minimal attack graph. Journal of Guangzhou University (Natural Science Edition), 2012, 11(3): 70–74 (in Chinese with English abstract). [doi: 10.3969/j.issn.1671-4229.2012.03.015]

[86] Cody T, Rahman A, Redino C, Huang LX, Clark R, Kakkar A, Kushwaha D, Park P, Beling P, Bowen E. Discovering exfiltration paths using reinforcement learning with attack graphs. In: Proc. of the 2022 IEEE Conf. on Dependable and Secure Computing (DSC). Edinburgh: IEEE, 2022. 1–8. [doi: 10.1109/DSC54232.2022.9888919]

[87] Ceri S, Gottlob G, Tanca L. What you always wanted to know about Datalog (and never dared to ask). IEEE Trans. on Knowledge and Data Engineering, 1989, 1(1): 146–166. [doi: 10.1109/69.43410]

[88] Ou XM, Govindavajhala S, Appel AW. MulVAL: A logic-based network security analyzer. In: Proc. of the 14th Conf. on USENIX Security Symp. Baltimore: USENIX Association, 2005. 113–128.

[89] Jajodia S, Noel S. Topological vulnerability analysis. In: Jajodia S, Liu P, Swarup V, Wang C, eds. Cyber Situational Awareness: Issues and Research. New York: Springer, 2009. 139–154. [doi: 10.1007/978-1-4419-0140-8_7]

[90] Artz ML. NetSPA: A network security planning architecture [MS. Thesis]. Cambridge: Massachusetts Institute of Technology, 2002.

[91] Ammann P, Wijesekera D, Kaushik S. Scalable, graph-based network vulnerability analysis. In: Proc. of the 9th ACM Conf. on

Computer and Communications Security. Washington: ACM, 2002. 217–224. [doi: 10.1145/586110.586140]
[92] Chen F, Liu DH, Zhang Y, Su JS. A scalable approach to analyzing network security using compact attack graphs. Journal of Networks, 2010, 5(5): 543–555.
[93] Wang LY, Yao C, Singhal A, Jajodia S. Interactive analysis of attack graphs using relational queries. In: Proc. of the 20th Data and Applications Security XX: 20th Annual IFIP WG 11.3 Working Conf. on Data and Applications Security. Sophia Antipolis: Springer, 2006. 119–132. [doi: 10.1007/11805588_9]
[94] Li W, Vaughn RB, Dandass YS. An approach to model network exploitations using exploitation graphs. Simulation, 2006, 82(8): 523–541. [doi: 10.1177/0037549706072046]
[95] Blum AL, Furst ML. Fast planning through planning graph analysis. Artificial Intelligence, 1997, 90(1–2): 281–300. [doi: 10.1016/S0004-3702(96)00047-1]
[96] Chu N, Chen XY, Zhang YF, Xin SY. Design and application of penetration attack tree model oriented to attack resistance test. In: Proc. of the 2008 Int'l Conf. on Computer Science and Software Engineering. Wuhan: IEEE, 2008. 622–626. [doi: 10.1109/CSSE.2008.1137]
[97] Kotenko I, Doynikova E. Security assessment of computer networks based on attack graphs and security events. In: Proc. of the 2nd IFIP TC5/8 Int'l Conf. on Information and Communication Technology. Bali: Springer, 2014. 462–471. [doi: 10.1007/978-3-642-55032-4_47]
[98] Luan JC, Wang J, Xue MF. Automated vulnerability modeling and verification for penetration testing using petri nets. In: Proc. of the 2nd Int'l Conf. on Cloud Computing and Security. Nanjing: Springer, 2016. 71–82. [doi: 10.1007/978-3-319-48674-1_7]
[99] Peterson JL. Petri nets. ACM Computing Surveys, 1977, 9(3): 223–252. [doi: 10.1145/356698.356702]
[100] Chowdhary A, Huang DJ, Mahendran JS, Romo D, Deng YL, Sabur A. Autonomous security analysis and penetration testing. In: Proc. of the 16th Int'l Conf. on Mobility, Sensing and Networking (MSN). Tokyo: IEEE, 2020. 508–515. [doi: 10.1109/MSN50589.2020.00086]
[101] Hoffmann J. Simulated penetration testing: From "Dijkstra" to "Turing Test++". Proc. of the Int'l Conf. on Automated Planning and Scheduling, 2015, 25(1): 364–372. [doi: 10.1609/icaps.v25i1.13684]
[102] Nau DS, Au TC, Ilghami O, Kuter U, Murdock JW, Wu D, Yaman F. SHOP2: An HTN planning system. Journal of Artificial Intelligence Research, 2003, 20: 379–404. [doi: 10.1613/jair.1141]
[103] Barrett A, Weld DS. Partial-order planning: Evaluating possible efficiency gains. Artificial Intelligence, 1994, 67(1): 71–112. [doi: 10.1016/0004-3702(94)90012-4]
[104] Coles A, Coles A, Fox M, Long D. Forward-chaining partial-order planning. Proc. of the Int'l Conf. on Automated Planning and Scheduling, 2021, 20(1): 42–49. [doi: 10.1609/icaps.v20i1.13403]
[105] De Silva L, Padgham L, Sardina S. HTN-like solutions for classical planning problems: An application to BDI agent systems. Theoretical Computer Science, 2019, 763: 12–37. [doi: 10.1016/j.tcs.2019.01.034]
[106] Ontañón S, Buro M. Adversarial hierarchical-task network planning for complex real-time games. In: Proc. of the 24th Int'l Joint Conf. on Artificial Intelligence. Buenos Aires: AAAI Press, 2015. 1652–1658.
[107] Mu CP, Li YJ. An intrusion response decision-making model based on hierarchical task network planning. Expert Systems with Applications, 2010, 37(3): 2465–2472. [doi: 10.1016/j.eswa.2009.07.079]
[108] Durkota K, Lisý V. Computing optimal policies for attack graphs with action failures and costs. In: Proc. of the 7th Starting AI Researchers' Symp. 2014. 101–110.
[109] Zhou TY, Zang YC, Zhu JH, Wang QX. NIG-AP: A new method for automated penetration testing. Frontiers of Information Technology & Electronic Engineering, 2019, 20(9): 1277–1288. [doi: 10.1631/FITEE.1800532]
[110] Tran K, Akella A, Standen M, Kim J, Bowman D, Richer T, Lin CT. Deep hierarchical reinforcement agents for automated penetration testing. arXiv:2109.06449, 2021.
[111] Ghanem MC, Chen TM. Reinforcement learning for intelligent penetration testing. In: Proc. of the 2nd World Conf. on Smart Trends in Systems, Security and Sustainability. London: IEEE, 2018. 185–192. [doi: 10.1109/WorldS4.2018.8611595]
[112] Ghanem MC, Chen TM. Reinforcement learning for efficient network penetration testing. Information, 2020, 11(1): 6. [doi: 10.3390/info11010006]
[113] Sarraute C, Buffet O, Hoffmann J. Penetration testing == POMDP solving? arXiv:1306.4714, 2013.
[114] Schwartz J, Kurniawati H, El-Mahassni E. POMDP + information-decay: Incorporating defender's behaviour in autonomous penetration testing. Proc. of the Int'l Conf. on Automated Planning and Scheduling, 2020, 30(1): 235–243. [doi: 10.1609/icaps.v30i1.6666]
[115] Common Vulnerability Scoring System v3.1: Specification document. 2023. https://www.first.org/cvss/v3.1/specification-document
[116] Smith B. Ontology. 2012. https://brill.com/display/book/9789401207799/B9789401207799-s005.xml

[117] Zhang BY, Wang M. Research on Quantization Method of Network Attack and Defense Based on CVSS Vulnerability Score. Journal of Ordnance Equipment Engineering, 2018, 39(4): 147–150.

[118] Figueroa-Lorenzo S, Añorga J, Arrizabalaga S. A survey of IIoT protocols: A measure of vulnerability risk analysis based on CVSS. ACM Computing Surveys, 2020, 53(2): 44. [doi: 10.1145/3381038]

[119] Duy Le T, Ge MM, The Duy P, Do Hoang H, Anwar A, Loke SW, Beuran R, Tan YS. CVSS based attack analysis using a graphical security model: Review and smart grid case study. In: Proc. of the 4th EAI Int'l Conf. on Smart Grid and Internet of Things. Taichung: Springer, 2021. 116–134. [doi: 10.1007/978-3-030-69514-9_11]

[120] Yadav T, Rao AM. Technical aspects of cyber Kill Chain. In: Proc. of the 3rd Int'l Symp. on Security in Computing and Communications. Kochi: Springer, 2015. 438–452. [doi: 10.1007/978-3-319-22915-7_40]

[121] CAPEC. 2023. https://capec.mitre.org/

[122] redcanaryco/atomic-red-team. 2023. https://github.com/redcanaryco/atomic-red-team

[123] Pinkston J, Undercofer J, Joshi A, Finin T. A target-centric ontology for intrusion detection. In: Proc. of the 18th Int'l Joint Conf. on Artificial Intelligence. 2003. 1–8.

[124] Herzog A, Shahmehri N, Duma C. An ontology of information security. Int'l Journal of Information Security and Privacy, 2007, 1(4): 1–23. [doi: 10.4018/jisp.2007100101]

[125] Wang JA, Guo MZ. OVM: An ontology for vulnerability management. In: Proc. of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies. Oak Ridge: ACM, 2009. 34. [doi: 10.1145/1558607.1558646]

[126] Gao JB, Zhang BW, Chen XH, Luo Z. Ontology-based model of network and computer attacks for security assessment. Journal of Shanghai Jiaotong University (Science), 2013, 18(5): 554–562. [doi: 10.1007/s12204-013-1439-5]

[127] Chu G, Lisitsa A. Ontology-based automation of penetration testing. In: Proc. of the 6th Int'l Conf. on Information Systems Security and Privacy. Valletta: Science and Technology Publications, 2020. 713–720. [doi: 10.5220/0009171007130720]

[128] Iannacone M, Bohn S, Nakamura G, Gerth J, Huffer K, Bridges R, Ferragut E, Goodall J. Developing an ontology for cyber security knowledge graphs. In: Proc. of the 10th Annual Cyber and Information Security Research Conf. Oak Ridge: ACM, 2015. 12. [doi: 10.1145/2746266.2746278]

[129] Ren YT, Xiao YJ, Zhou YH, Zhang ZY, Tian ZH. CSKG4APT: A cybersecurity knowledge graph for advanced persistent threat organization attribution. IEEE Trans. on Knowledge and Data Engineering, 2023, 35(6): 5695–5709. [doi: 10.1109/TKDE.2022.3175719]

[130] Belief-desire-intention software model. 2023. https://en.wikipedia.org/wiki/Belief%E2%80%93desire%E2%80%93intention_software_model

[131] Gao Y, Chen SF, Lu X. Research on reinforcement learning technology: A review. Acta Automatica Sinica, 2004, 30(1): 86–100 (in Chinese with English abstract). [doi: 10.16383/j.aas.2004.01.011]

[132] gyoisamurai/gyoithon. 2023. https://github.com/gyoisamurai/GyoiThon

[133] Greenwald L, Shanley R. Automated planning for remote penetration testing. In: Proc. of the 2009 IEEE Military Communications Conf. Boston: IEEE, 2009. 1–7. [doi: 10.1109/MILCOM.2009.5379852]

[134] Myasnikov AV, Konoplev AS, Suprun AF, Anisimov VG, Kasatkin VV, Los' VP. Constructing the model of an information system for the automatization of penetration testing. Automatic Control and Computer Sciences, Springer, 2021, 55(8): 949–955.

[135] Dulac-Arnold G, Evans R, Van Hasselt H, Sunehag P, Lillicrap T, Hunt J, Mann T, Weber T, Degris T, Coppin B. Deep reinforcement learning in large discrete action spaces. arXiv:1512.07679, 2016.

[136] Zahavy T, Haroush M, Merlis N, Mankowitz DJ, Mannor S. Learn what not to learn: Action elimination with deep reinforcement learning. arXiv:1809.02121, 2019.

[137] Zhou SC, Liu JJ, Hou DD, Zhong XF, Zhang Y. Autonomous penetration testing based on improved deep Q-network. Applied Sciences, 2021, 11(19): 8823. [doi: 10.3390/app11198823]

[138] Maeda R, Mimura M. Automating post-exploitation with deep reinforcement learning. Computers & Security, 2021, 100: 102108. [doi: 10.1016/j.cose.2020.102108]

[139] Advantage actor critic. 2023. https://huggingface.co/blog/deep-rl-a2c

[140] Q-learning. 2023. https://en.wikipedia.org/wiki/Q-learning

[141] State-action-reward-state-action. 2023. https://en.wikipedia.org/wiki/State%E2%80%93action%E2%80%93reward%E2%80%93state%E2%80%93action

[142] Niakanlahiji A, Wei JP, Alam MR, Wang QY, Chu BT. Shadowmove: A stealthy lateral movement strategy. In: Proc. of the 29th

USENIX Conf. on Security Symp. 2020. 559–576.
[143] Chaudhary S, O'Brien A, Xu SJ. Automated post-breach penetration testing through reinforcement learning. In: Proc. of the 2020 IEEE Conf. on Communications and Network Security (CNS). Avignon: IEEE, 2020. 1–2. [doi: 10.1109/CNS48642.2020.9162301]
[144] 13o-bbr-bbq/machine_learning_security. 2023. https://github.com/13o-bbr-bbq/machine_learning_security
[145] crond-jaist/autopentest-drl. 2023. https://github.com/crond-jaist/AutoPentest-DRL
[146] OpenVAS. 2023. https://www.openvas.org/
[147] Qian KX, Zhang DJ, Zhang P, Zhou ZH, Chen XZ, Duan SX. Ontology and reinforcement learning based intelligent agent automatic penetration test. In: Proc. of the 2021 IEEE Int'l Conf. on Artificial Intelligence and Computer Applications (ICAICA). Dalian: IEEE, 2021. 556–561. [doi: 10.1109/ICAICA52286.2021.9497911]
[148] Semantic Web rule language. 2023. https://en.wikipedia.org/wiki/Semantic_Web_Rule_Language
[149] Protégé. 2023. https://protege.stanford.edu/
[150] Schwartz J, Kurniawati H. Autonomous penetration testing using reinforcement learning. arXiv:1905.05965, 2019.
[151] Almubairik NA, Wills G. Automated penetration testing based on a threat model. In: Proc. of the 11th Int'l Conf. for Internet Technology and Secured Transactions (ICITST). Barcelona: IEEE, 2016. 413–414. [doi: 10.1109/ICITST.2016.7856742]
[152] Applebaum A, Miller D, Strom B, Korban C, Wolf R. Intelligent, automated red team emulation. In: Proc. of the 32nd Annual Conf. on Computer Security Applications. Los Angeles: ACM, 2016. 363–373. [doi: 10.1145/2991079.2991111]
[153] Chu G, Lisitsa A. Poster: Agent-based (BDI) modeling for automation of penetration testing. In: Proc. of the 16th Annual Conf. on Privacy, Security and Trust (PST). Belfast: IEEE, 2018. 1–2. [doi: 10.1109/PST.2018.8514211]
[154] Jiao J, Zhao HN, Cao HS. Using deep learning to construct auto Web penetration test. In: Proc. of the 13th Int'l Conf. on Machine Learning and Computing. Shenzhen: ACM, 2021. 59–66. [doi: 10.1145/3457682.3457691]
[155] Yang YZ, Liu X. Behaviour-diverse automatic penetration testing: A curiosity-driven multi-objective deep reinforcement learning approach. arXiv:2202.10630, 2022.
[156] Auricchio N, Cappuccio A, Caturano F, Perrone G, Romano SP. An automated approach to Web offensive security. Computer Communications, 2022, 195: 248–261. [doi: 10.1016/j.comcom.2022.08.018]
[157] Chen JY, Hu SL, Zheng HB, Xing CY, Zhang GM. GAIL-PT: A generic intelligent penetration testing framework with generative adversarial imitation learning. arXiv:2204.01975, 2022.
[158] Xu DX, Tu MH, Sanford M, Thomas L, Woodraska D, Xu WF. Automated security test generation with formal threat models. IEEE Trans. on Dependable and Secure Computing, 2012, 9(4): 526–540. [doi: 10.1109/TDSC.2012.24]

附中文参考文献:

[4] 王欣, 桂畅旎. 2022 上半年网络安全漏洞态势观察. 中国信息安全, 2022(9): 85–87. [doi: 10.3969/j.issn.1674-7844.2022.09.029]
[9] 中华人民共和国网络安全法. 2016. http://www.cac.gov.cn/2016-11/07/c_1119867116.htm
[10] 关键信息基础设施安全保护条例 (中华人民共和国国务院令第 745 号). 2021. http://www.gov.cn/zhengce/content/2021-08/17/content_5631671.htm
[34] 严俊龙. 基于 Metasploit 框架自动化渗透测试研究. 信息网络安全, 2013(2): 53–56. [doi: 10.3969/j.issn.1671-1122.2013.02.014]
[35] 卜佑军, 王涵, 胡静萍, 张桥. 一种自动化安全渗透测试系统的设计与研究. 网络安全技术与应用, 2020(7): 32–34. [doi: 10.3969/j.issn.1009-6833.2020.07.022]
[39] 葛昕, 岳敏楠, 金江涛. 基于校园网的自动化渗透测试框架研究. 深圳大学学报 (理工版), 2020, 37(S1): 68–72. [doi: 10.3724/SP.J.1249.2020.99068]
[40] 陈思琪. 基于自动化渗透性测试的军交网络安全研究 [硕士学位论文]. 兰州: 兰州大学, 2012.
[51] 王持恒, 陈晶, 苏涵, 何琨, 杜瑞颖. 基于宿主权限的移动广告漏洞攻击技术. 软件学报, 2018, 29(5): 1392–1409. http://www.jos.org.cn/1000-9825/5494.htm [doi: 10.13328/j.cnki.jos.005494]
[57] 成瑜. 攻击载荷自动化生成系统的研究与实现 [硕士学位论文]. 西安: 西安电子科技大学, 2017.
[71] 闻观行, 张园超, 张玉清. 基于数据格式支持机制的自动化渗透测试框架. 中国科学院研究生院学报, 2011, 28(5): 676–683.
[73] 高宏佳, 李世明. 基于自动化的渗透测试. 智能计算机与应用, 2020, 10(4): 162–164. [doi: 10.3969/j.issn.2095-2163.2020.04.040]
[74] 邢斌, 高岭, 孙骞, 杨威. 一种自动化的渗透测试系统的设计与实现. 计算机应用研究, 2010, 27(4): 1384–1387. [doi: 10.3969/j.issn.1001-3695.2010.04.048]
[76] 卢继军, 黄刘生, 吴树峰. 基于攻击树的网络攻击建模方法. 计算机工程与应用, 2003, 39(27): 160–163. [doi: 10.3321/j.issn:1002-8331.2003.27.051]

[80] 陈锋, 张怡, 苏金树, 韩文报. 攻击图的两种形式化分析. 软件学报, 2010, 21(4): 838–848. http://www.jos.org.cn/1000-9825/3584.htm [doi: 10.3724/SP.J.1001.2010.03584]

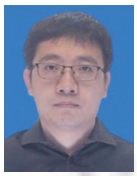
[82] 王国玉, 王会梅, 陈志杰, 鲜明. 基于攻击图的计算机网络攻击建模方法. 国防科技大学学报, 2009, 31(4): 74–80. [doi: 10.3969/j.issn.1001-2486.2009.04.015]

[85] 谢冬青, 李贵城. 基于最小化攻击图的自动化渗透测试模型. 广州大学学报 (自然科学版), 2012, 11(3): 70–74. [doi: 10.3969/j.issn.1671-4229.2012.03.015]

[117] 张必彦, 王孟. 基于 CVSS 漏洞评分标准的网络攻防量化方法研究. 兵器装备工程学报, 2018, 39(4): 147–150.

[131] 高阳, 陈世福, 陆鑫. 强化学习研究综述. 自动化学报, 2004, 30(1): 86–100. [doi: 10.16383/j.aas.2004.01.011]

陈可(1998－), 男, 硕士, 主要研究领域为网络攻防对抗, 自动渗透技术.



孙彦斌(1987－), 男, 博士, 教授, CCF 高级会员, 主要研究领域为网络安全, 工控系统安全, 未来网络.

鲁辉(1981－), 男, 博士, 教授, 博士生导师, CCF 专业会员, 主要研究领域为网络攻防对抗, 智能化漏洞挖掘, 移动端脱壳和反混淆技术.

苏申(1985－), 男, 博士, 教授, CCF 专业会员, 主要研究领域为区块链安全, DNS 安全, 路由安全, 车联网安全.

方滨兴(1960－), 男, 博士, 教授, 博士生导师, 主要研究领域为计算机网络, 信息安全.

田志宏(1978－), 男, 博士, 教授, 博士生导师, CCF 杰出会员, 主要研究领域为网络攻防对抗, 漏洞挖掘与利用, APT 检测与溯源, 工控安全.