引用格式：陆鹏程，钟晓峰，陈杰，等.“衔尾蛇”：基于符号增强网络与深度强化学习的自动化 WAF 安全测试框架[J]. 信息对抗技术，2025，4(5)：66-76.[LU Pengcheng，ZHONG Xiaofeng，CHEN Jie，et al. “Ourboros”：an automated WAF security testing framework based on symbol-enhanced networks and deep reinforcement learning[J]. Information Countermeasure Technology，2025，4(5)：66-76.(in Chinese)]

“衔尾蛇”：基于符号增强网络与深度强化学习的自动化 WAF 安全测试框架

陆鹏程[1,2]，钟晓峰[1,2]，陈　杰[1,2]，许文博[1,2]，王永杰[1,2*]

（1. 国防科技大学电子对抗学院，安徽合肥 230037；2. 网络空间安全态势感知与评估安徽省重点实验室，安徽合肥 230037）

摘　要　Web 应用防火墙（Web application firewall，WAF）是应对持续性威胁的关键防御机制，但其安全评估长期面临挑战。传统人工测试方法效率低下且资源耗费大，而现有基于强化学习（reinforcement learning，RL）的自动化方案存在两大局限：一是攻击者因无法感知 WAF 的不透明规则逻辑，导致黑盒测试效率低下；二是 WAF 的布尔值反馈引发稀疏/延迟奖励问题，稀疏奖励易使智能体陷入盲目探索，延迟奖励则阻碍早期操作与最终结果的关联，严重影响学习效率。为突破上述瓶颈，首次提出“衔尾蛇”——黑盒 WAF 测试框架，其核心在于将提取的 WAF 规则转化为可解释循环神经网络（recurrent neural network，RNN），以提供细粒度置信度评分，并融合该评分与最终结果级奖励来驱动强化学习测试。实验表明，该框架在基于特征的 WAF 上最高可实现 89.2%的规避成功率，这不仅缓解稀疏奖励问题，提供了高效的黑盒测试方案，还为优化 WAF 规则提供了重要参考。

关键词　深度强化学习；正则表达式；SQL 注入；WAF 安全测试

中图分类号　TP 309　　**文章编号**　2097-163X(2025)05-0066-11

文献标志码　A　　**DOI**　10.12399/j.issn.2097-163x.2025.05.005

“Ourboros”：an automated WAF security testing framework based on symbol-enhanced networks and deep reinforcement learning

LU Pengcheng[1,2]，ZHONG Xiaofeng[1,2]，CHEN Jie[1,2]，XU Wenbo[1,2]，WANG Yongjie[1,2*]

（1. College of Electronic Engineering，National University of Defense Technology，Hefei 230037，China；
2. Anhui Province Key Laboratory of Cyberspace Security Situation Awareness and Evaluation，Hefei 230037，China）

Abstract　Web application firewall（WAF）is critical defensive mechanisms against persistent threats，yet its security assessment has long been challenging. Traditional manual testing methods are inefficient and resource-intensive，while existing reinforcement learning（RL）based methods suffer from two major limitations：first，attackers cannot perceive the opaque rule logic of WAF，leading to low efficiency in black-box testing；second，the Boolean feedback of WAF causes the problem of sparse/delayed rewards—sparse rewards tend to trap intelligent agents in blind exploration，and delayed rewards hinder the association between early actions and final outcomes，seriously impairing learning efficiency. To break through

收稿日期：2025-07-11　　**修回日期**：2025-08-20

通信作者：王永杰，E-mail：wangyongjie17@nudt.edu.cn

基金项目：国家自然科学基金资助项目（62271496）；军队装备综合研究项目（KY24N011）

these bottlenecks, this study proposed "Ouroboros"—a black-box WAF testing framework—for the first time. Its core lies in converting the extracted WAF rules into an interpretable recurrent neural network (RNN) to provide fine-grained confidence scores, and integrating these scores with outcome-level rewards to drive RL-based testing. Experiments show that this framework can achieve a maximum bypass success rate of 89.2% on feature-based WAF. This not only alleviates the sparse reward problem and provides an efficient black-box testing solution, but also offers important references for optimizing WAF rules.

Keywords　deep reinforcement learning; regular expression; SQL injection; WAF security testing

0　引言

Web 应用防火墙(Web application firewall, WAF)是应对持续性威胁的关键防御机制。然而，不断增长的网络攻击威胁、WAF 自身设计缺陷以及精心构造的高级恶意载荷，正持续挑战 WAF 的防御能力[1]，这使针对 WAF 安全测试(尤其是自动化测试技术)的研究愈发关键。根据规避 WAF 的漏洞机制，可将其分为载荷级规避与协议级规避 2 类。协议级规避通过分析 WAF 与源服务器解析 HTTP 请求时的语义差异，或利用内容分发网络(content delivery network, CDN)处理 HTTP 请求时对 RFC 标准支持的不一致性实现，旨在协议层面突破 WAF 防护[2-4]。载荷级规避则通过利用不完善的 WAF 规则过滤机制、配置逻辑缺陷以及攻击载荷的多态性特征，对原始攻击载荷实施语义保持的转换以规避检测[5-9]。本研究聚焦于载荷级黑盒自动化规避测试，旨在发现防护规则漏洞，其核心挑战在于如何高效地生成既能规避检测又保持攻击语义的变异载荷。

当前，载荷级规避技术主要分为基于搜索、基于变异和基于生成 3 类方法。基于搜索的方法(如 RAT 工具[10])通过 n-gram 分词聚类相似载荷，采用带ϵ-贪婪策略的强化学习(reinforcement learning, RL)进行自适应探索。基于变异的方法依赖转换/混淆技术生成载荷变体，典型案例如 WAF-A-MoLE[11]建立优先级队列系统、AdvSQLi[12]将 SQL 注入载荷映射为抽象语法树并通过上下文无关文法生成变体以及 ML-driven[5]利用遗传算法结合随机森林预测的演化框架。此外，YAO 等[6]采用深度 RL(deep RL, DRL)扰动载荷，以 WAF 分类器分数作为奖励信号；HEMMTAI 等[13]通过随机网络蒸馏将该方法扩展至黑盒场景。生成式方法代表包括 CHOWDHARY 等[14]开发的基于语义分词的条件序列生成对抗网络(generative adversarial network, GAN)、GPTfuzzer[15]融合上下文无关文法与大语言模型(large language model, LLM)微调的技术路线以及 XploitSQL[7]采用"演员-评论家"架构微调 T5 模型生成定向 SQL 注入载荷的框架。需指出，现有技术存在着以下显著局限：搜索方法受限于数据集规模与局部最优陷阱；生成方法面临 GAN 的语义失真与 LLM 的幻觉问题；变异方法虽平衡语义保持与多样性，但过度依赖 WAF 反馈机制。

在使用强化学习对 WAF 进行黑盒测试时，因仅能从 WAF 获得布尔型反馈，稀疏奖励(仅当载荷完全规避 WAF 时获得正反馈)与延迟奖励(复杂攻击路径的成功依赖多步关键变异，但奖励仅在终点分配)问题严重制约学习效率，导致探索过程陷入盲目搜索。

为此，本文提出"衔尾蛇"——一个黑盒 WAF 自动化测试框架，将提取的 WAF 规则转化为可解释循环神经网络(recurrent neural network, RNN)以提供细粒度置信度评分，并融合此评分与结果级奖励驱动强化学习对 WAF 进行安全测试。实验表明，该框架能够有效对 WAF 规则进行窃取，克隆 WAF 的准确率达到了原有 WAF 的 85%。载荷变形阶段，在基于特征的 WAF 上达到了最高 89.2%的规避成功率。这不仅提供了高效的黑盒自动化测试方案，缓解了稀疏奖励问题，也为优化 WAF 规则提供了重要参考。

本文的主要贡献如下：

1) 新型框架架构。提出首个融合符号规则提取、规则神经网络化与 DRL 测试的闭环自增强框架——"衔尾蛇"，实现自动化黑盒 WAF 测试。

2) 突破稀疏/延迟奖励瓶颈。通过将黑盒规

则转化为输出细粒度置信度评分的 RNN 模型，提供丰富的中间信号；融合该评分与最终规避结果构建稠密奖励机制，有效缓解基于 RL 的 WAF 测试中稀疏/延迟奖励的核心难题。

3）高效规避能力验证。实验证明框架具备良好的规避能力，在基于特征的 WAF 上达到 89.2%的峰值规避成功率，在训练效率与成功率上显著优于现有 RL 方案。

1 威胁建模

1.1 目标系统设定

本文提出的框架面向基于签名的 WAF，基于签名的防火墙可分为基于正则表达式的过滤方式以及基于语义的过滤方式。基于正则表达式的检测引擎通过设置一组预定义的正则式对 HTTP 请求中待检查的参数进行匹配。而基于语义分析的引擎首先根据参数的语义将其转换为指纹，通过二分匹配预定义的指纹库来进行检测。本文的攻击目标设定为其中基于正则表达式引擎 WAF，RL 则是驱动攻击的方法。

1.2 攻击者能力

测试者已知 WAF 的类型，但对其规则、规则数量和内部执行情况未知，仅能根据 HTTP 请求的状态返回码获得能否规避的反馈。测试者可无限制地向 WAF 发送 HTTP 请求并保留日志，日志会记录从发送 HTTP 请求到接受 WAF 反馈的时间。现有 WAF 规避方法分为载荷层规避与协议层规避，二者区别如图 1 所示。测试者仅能对 http 请求中的参数部分（即载荷）进行变形，无法改变协议头中的信息。

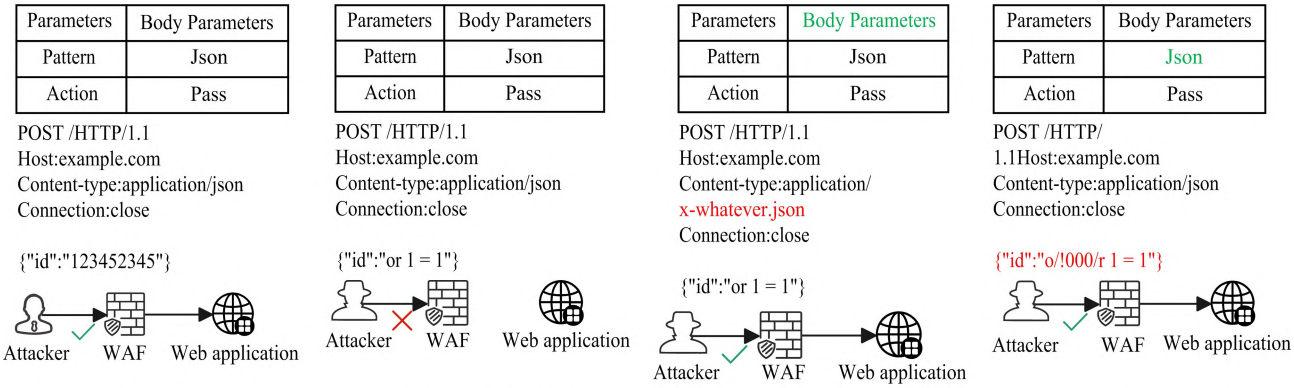


图 1 载荷层规避与协议层规避的不同

Fig. 1 The differences between payload-level evasion and protocol-level evasion

2 方法设计

“衔尾蛇”总体框架如图 2 所示，主要分为 3 个部分：基于遗传算法的 WAF 规则提取、符号增强网络生成和基于 RL 的载荷变异。其核心思想在于为克服 RL 在黑盒环境下存在的稀疏奖励问题，尝试提取黑盒 WAF 的检测规则，将黑盒攻击转换为白盒攻击。然而，提取的正则表达式规则仍然是布尔型反馈，未能解决稀疏奖励问题。究其原因，符号规则具备可解释性，却难以被程序直接利用。为此，通过将符号规则进行神经网络化处理，以输出反映载荷恶意检出可能性的奖励信号，持续指导 RL 模型的训练。RL 训练会产生大量失败的中间数据，本框架能够利用这些中间数据进行自增强，将中间数据重新进行基于遗传算法的 WAF 规则提取，以获取更加精确的规则，进而促进变异载荷的生成，“衔尾蛇”的命名正是源于这种自循环的设计理念。

2.1 基于遗传算法的 WAF 规则提取

基于遗传算法的 WAF 规则提取如图 3 所示。已有的研究未能充分利用与 WAF 交互后被拦截的有效载荷，这些数据隐含了 WAF 正则表达式的过滤逻辑。从被拦截载荷中提取共有模式可逼近实际 WAF 规则的子集，其精度随数据的积累而提升。以下从预处理与基因初始化、基因编码与解码机制和适应度评估 3 个方面介绍。

2.1.1 预处理与基因初始化

在预处理阶段，首先，从 WAF 拦截的恶意载荷中提取关键匹配模式。对于样本中出现的 SQL 固定结构（如"SELECT""UNION"等），保

留其原始形式，避免因过度泛化导致规则覆盖范围超出实际需求。针对每个载荷，采用端点腐蚀法提取最小匹配单元：从两端逐字符移除直至匹配条件失效，确定核心特征子串后，将非关键部分替换为通配符（例如载荷"admin’OR 1＝1/＊"经处理得到模式"OR 1＝1"）。通过 TF-IDF 嵌入，将文本信息转化为多维空间中的向量，这些向量能够反映载荷间的相似性与差异性。随后，利用基于密度的空间聚类（density-based spatial clustering of applications with noise，DBSCAN）算法对转换后的载荷向量进行聚类处理。该算法能在噪声环境下识别任意形态的簇，即计算各点在给定半径 ρ 内的邻近点数（即点密度），将高密度区域划分为簇，低密度点标记为噪声，这样既无需预先指定聚类数量，又能自动将载荷划分为若干具有相似匹配特征的同构类簇。此类分组操作不仅提升了数据处理效率，更显著降低了后续正则表达式生成的复杂度。

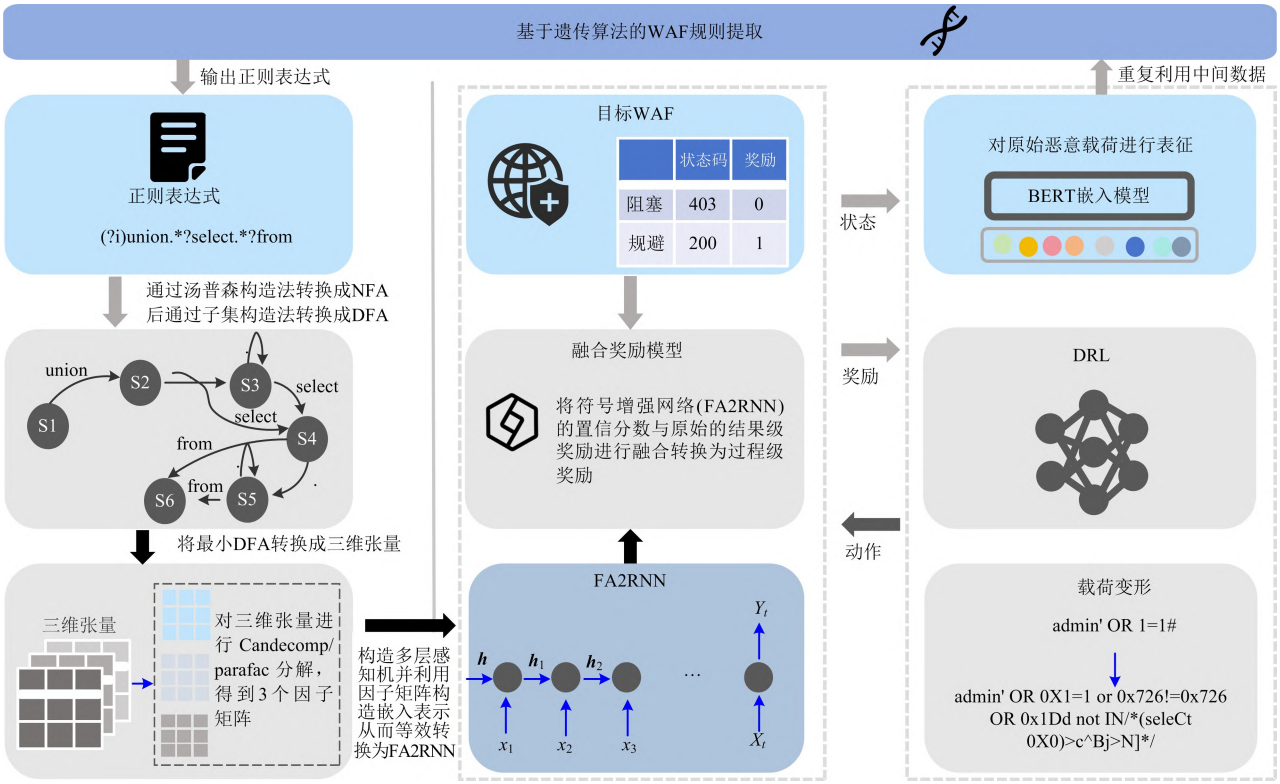


图 2　“衔尾蛇”总体框架

Fig. 2　The overall framework of Ourboros

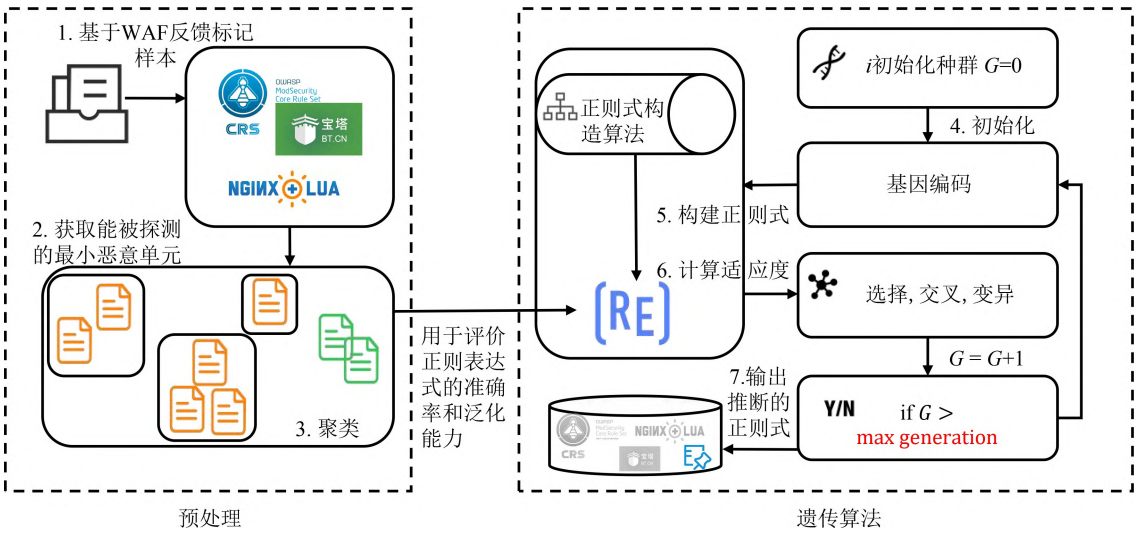


图 3　基于遗传算法的 WAF 规则提取

Fig. 3　Genetic algorithm-based WAF rule extraction

2.1.2 基因编码与解码机制

基因型为十六进制序列，每个基因对应正则表达式的特定组件。编码阶段遵循优先级编码原则(从左向右解析载荷，高位基因优先匹配)，例如，"abc1"可通过基因 0x620 和 0x206 分别表示为"6666(\w\w\w\w)"与"2220([a-z][a-z][a-z]\d)"。解码阶段首先将基因映射后的载荷转换为中间(值-长度)表示形式，并通过填充(*，n)实现载荷序列等长化。以"6666(\w\w\w\w)"为例，其对应表示为"(6,4)"。基于此中间态构建动态规划矩阵，通过状态转移方程求解最长公共子序列(longest common subsequence，LCS)。当矩阵回溯路径完成时，即可生成覆盖所有载荷的共享正则表达式(表型)。通过多尺度泛化策略，对表型序列的非 LCS 部分进行列向分析(包含字母数字的列泛化为\w，纯字母列转为[A-Z a-z])，最终合并 LCS 内核与泛化结果，构造候选正则表达式。

2.1.3 适应度评估机制

为平衡生成规则的准确性与泛化能力，本文设计了复合适应度函数。对误判样本(正例判为负例/负例判为正例)施加高权重惩罚以保持规则有效性，同时，通过正则表达式长度惩罚项与字符集复杂度指标控制泛化。

2.2 符号增强网络生成

有限状态自动机(finite state automaton，FSA)是描述具有有限离散状态系统行为的数学模型，其状态转换由特定的输入触发。通过汤普森构造算法[16]可以将正则表达式转换为 FSA，再经确定型有限状态自动机(deterministic finite automaton，DFA)构造算法[17]与 DFA 最小化算法[18]处理，可为给定正则表达式生成唯一的 DFA。在正则表达式转换前，首先收集 SQL 注入载荷中的关键词进行分词处理，并将关键词集合定义为 V_{keyword}，ASCII 字符集有效字符定义为 V_{char}。DFA 形式化为五元组 $A=(\Sigma,S,\boldsymbol{T},\boldsymbol{\alpha}_0,\boldsymbol{\alpha}_\infty)$，其中：$\Sigma$ 为输入词汇表，本框架中 $|\Sigma|=|V_{\text{keyword}}|+|V_{\text{char}}|$；$S$ 为有限状态集合，$|S|=K$；$\boldsymbol{T}\in\mathbf{R}^{|\Sigma|\times K\times K}$ 为状态转移权重张量。$\boldsymbol{T}[\sigma,S_i,S_j]$ 表示输入 σ，状态 S_i 能否转移到状态 S_j，DFA 中仅取 0(不可转移)或 1(可转移)；$\boldsymbol{\alpha}_0\in\mathbf{R}^K$ 为初始状态权重向量，$\boldsymbol{\alpha}_0[i]$ 表示 $t=0$ 时刻状态 S_i 的激活权重；$\boldsymbol{\alpha}_\infty\in\mathbf{R}^K$ 为终止状态权重向量，$\boldsymbol{\alpha}_\infty[i]$ 表示完整读取输入后状态 S_i 的终止权重。

对于输入序列 $X=\{x_1,x_2,\cdots,x_n\}$ 及状态路径 $p=\{U_1,U_2,\cdots,U_{n+1}\}$($U_t$ 表示 t 时刻状态)，路径评分 $B(A,p)$ 定义为：

$$B(A,p)=\boldsymbol{\alpha}_0[u_1]\cdot\left(\prod_{i=1}^{N}T[\sigma,u_i,u_{i+1}]\right)\cdot\boldsymbol{\alpha}_\infty[u_{N+1}] \tag{1}$$

式(1)量化了输入序列沿特定状态路径被 FSA 接受的整体可能性。它通过将路径起点(初始状态权重)、路径中每一步的状态转移(转移张量元素的连乘)和路径终点(终止状态权重)的概率(或 DFA 中的 0/1 指示)相乘，得到一个综合评分。该评分代表了序列遵循路径完成状态转移并被最终状态接受的可能性。路径评分为 1 代表该序列能够到达最终状态，也就说明了该序列能被该自动机和该自动机对应的正则表达式接受，即 WAF 规则能够检测出这条 SQL 注入载荷。

这种自动机结构与 RNN 存在本质相似性(二者均在时刻 $t+1$ 接受输入信息，结合 t 时刻的隐藏状态生成 $t+1$ 时刻的隐藏状态)，因此，加权有限自动机(weighted finite automaton，WFA)的推导过程可重构为循环形式。该模型计算输入序列 x 处理完前 t 个词后的前向分数向量 $\boldsymbol{h}_t\in\mathbf{R}^K$($K$ 表示 WFA 状态总数)，其分量 $\boldsymbol{h}_t[i]$ 表示消耗 t 个词后可达状态 i 的概率。

自动机与 RNN 的等价性已在文献[19]中得到证实，研究者成功从 RNN 隐藏态中提取出自动机状态。文献[20]进一步揭示了 RNN 状态与最小化确定型有限自动机(minimization DFA，MDFA)超状态的映射关系，实现用 RNN 模拟自动机行为。这种等价关系是双向的——RNN 可视为参数化加权自动机。本文通过加权自动机，在有限自动机(取值 0/1)与 RNN 之间建立桥梁：从神经网络视角看，其为线性激活的循环网络；从自动机视角看，其为有限状态机。该架构兼具参数可更新性与高度可解释性，其参数更新过程可理解为搜索匹配状态转移的最优自动机(RNN 隐藏态对应加权自动机状态，具有正则表达式匹配当前输入的物理意义[21])。由于自动机参数量远超传统 RNN，本文采用 CANDECOMP/PARAFAC 张量分解(CPD)实现轻量化。CPD 将高阶张量分解为若干秩 1 因子张量的和，设三阶张量 $\boldsymbol{X}\in\mathbf{R}^{I\times J\times K}$，其分解式为：

$$\hat{\boldsymbol{X}}=\sum_{r=1}^{R}\lambda_r a_r\circ b_r\circ c_r=\left[\left[\lambda;A,B,C\right]\right] \tag{2}$$

式中，"∘"表示向量外积，张量分解秩 R 为超参数，重构因子矩阵张量记为 $\boldsymbol{X}$。此时，张量分解问题转化为最小化重构误差问题：

$$\min_{\hat{\boldsymbol{X}}} \| \boldsymbol{X} - \hat{\boldsymbol{X}} \|^2 \tag{3}$$

将转移张量 $\boldsymbol{T}$ 分解为 3 个因子矩阵 $\boldsymbol{E}_r \in \mathbf{R}^{|\Sigma| \times R}$，$\boldsymbol{D}_1 \in \mathbf{R}^{K \times R}$，$\boldsymbol{D}_2 \in \mathbf{R}^{K \times R}$，其中，$\boldsymbol{E}_r$ 可视为融合正则表达式信息的词向量嵌入矩阵，输入嵌入矩阵得到的嵌入向量的每一维代表了自动机的一个终止状态。例如，嵌入向量[0.87，0.01，0.46，…]，0.87 代表了载荷到达自动机的第 1 个终止状态，即被其中一个正则表达式匹配的概率为 0.87。式(1)为使用自动机张量时的路径评分函数，自动机张量在张量分解后表示为：

$$\boldsymbol{h}_t = ((\boldsymbol{h}_{t-1} \cdot \boldsymbol{D}_1) \odot v_t) \cdot \boldsymbol{D}_2^{\mathrm{T}} \tag{4}$$

式中，⊙表示哈达玛积。

RNN 生成的嵌入向量(根据更新后的路径评分函数通过矩阵运算获得)并非恶意载荷最终被检出的概率，而是需多层感知机(multilayer perceptron，MLP)进一步处理的多维特征向量。MLP 通过融合特征实现分类判定，实现标签概率估计。依据通用逼近定理[22]，MLP 通过调整参数可逼近任意连续函数，其本质是构建分类时特征间的逻辑关系。

2.3　基于 DRL 的扰动决策模型

本文将 WAF 规避问题的分析建模为部分可观测的马尔可夫决策过程。

2.3.1　状态空间设计

使用 BERT 嵌入对由攻击载荷和 WAF 反馈组成的状态进行向量化操作：通过 WordPiece 对载荷进行分词生成(包含全词、子词及特殊标记)的序列，融合词嵌入(语义特征)、位置嵌入(序列关系)、分段嵌入(语义边界)形成复合表征，再经 Transformer 多层自注意力机制处理，输出具有上下文感知的语义编码，提升分析复杂恶意载荷的能力。

2.3.2　动作空间设计

动作空间设计见表 1 所列，其中包含 33 种在保持查询语义前提下改变载荷结构的变异算子。该集合通过整合 SQLmap 的 tamper 脚本及上下文无关文法(CFG)生成的等效替换实现(原始算子 8 个＋新增 25 个)。

表 1　部分动作的对应表

Tab. 1　Correspondence table for a subset of actions

动作	示例
Space_to_comments	admin or 1＝1→admin/＊＊/or/＊＊/1＝1
Random_case	admin or 1＝1→admin or 1＝1
Logical_invariant	admin or 1＝1→admin or 1＝1 and True
Swap_keywords	admin or 1＝1→admin || 1＝1
Swap_int_repr	admin or 1＝1→admin or 0x1＝1
Comment_rewriting	admin or 1＝1→admin or 1/＊abc＊/＝1
Change_tautologies	admin or 1＝1→admin or 2<>3

CFG 属于乔姆斯基 2 型文法，其形式化定义为四元组 $G=(S,V,\Sigma,R)$：S 为起始符号集；V 为非终结符集；Σ 为终结符集(不可由规则生成)；R 为产生式规则集(左部＝头部，右部＝体部)。

具体推导规则与示例如图 4 所示。

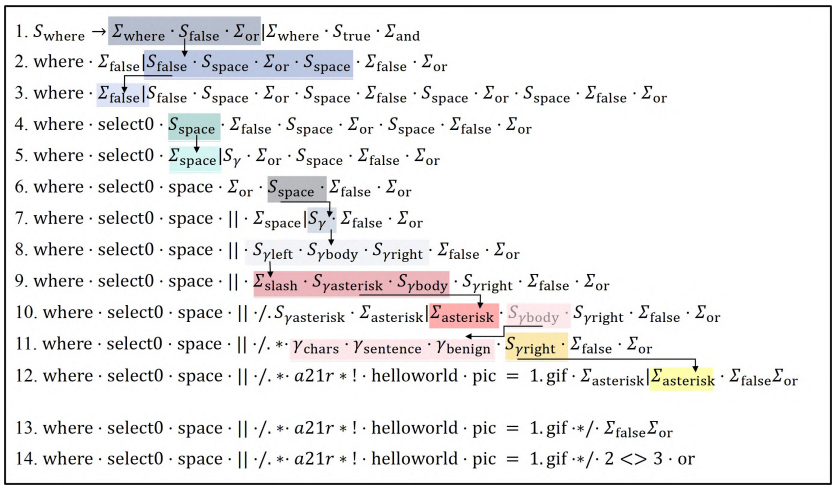


图 4　利用上下文无关文法生成替换项的推导过程

Fig. 4　Derivation process of context-free grammar

2.3.3 奖励函数设计

原始的结果级奖励根据载荷能否规避目标WAF检测来判断是否完成任务，evade表示成功，block表示失败，其表达式为：

$$R_{\text{sys}}=\begin{cases}10, & \text{evade}\\ 0, & \text{block}\end{cases} \tag{5}$$

式(5)定义了RL智能体从目标环境WAF获得的基础奖励信号，其为二元稀疏奖励函数：若智能体生成的变异载荷成功规避WAF的检测(evade)，则获得较大正向奖励(+10)；若载荷被WAF拦截(block)，则获得零奖励(0)。该奖励直接反映智能体动作序列(载荷变异过程)的最终成败结果。采用式(5)是因其能最直接地反映了RL任务(规避WAF)的终极目标，其模拟了黑盒WAF测试场景中攻击者实际能获得的反馈类型——通常只能得知请求是被放行还是被拦截(布尔值结果)。该奖励是任务驱动的核心信号，然而，其稀疏性(仅在完全成功时获得)和延迟性(成功可能依赖多步关键变异，奖励却仅在终点分配)是导致传统RL方法在WAF测试中效率低下的关键原因(详见引言)，这也凸显了引入额外过程奖励的必要性。

为此，将变形前载荷 χ_{orig} 与变形后载荷 χ_{mut} 分别输入符号增强网络FA2RNN，输出的置信分数差值作为过程奖励，表达式为：

$$R_{\text{reg}}=\text{FA2RNN}(\chi_{\text{orig}})-\text{FA2RNN}(\chi_{\text{mut}}) \tag{6}$$

其实际含义为计算单次变异操作引起的置信度变化，用于提供细粒度、即时的过程奖励，指导智能体学习每一步变异的效果。采用式(6)的核心目的是解决式(5)所存在的稀疏和延迟奖励问题。FA2RNN作为由克隆规则转化的神经网络，提供了对载荷的细粒度、即时评估。R_{reg} 量化了每一次变异动作对规避规则效果的直接影响(即使载荷最终尚未完全规避)，为RL智能体提供了丰富、密集的中间学习信号，指导智能体理解哪些变异操作(动作)是有效的(降低检出风险)，哪些是无效的，从而显著加速探索和学习过程，避免在无效路径上浪费预算，有效缓解了稀疏/延迟奖励困境。

最终的合成奖励由式(5)和式(6)融合得到。同时对其范围进行约束，表达式为：

$$R_{\text{sin}}=\min\left(10,\max\left(0,\sum_{k\in\{\text{sys},\text{reg}\}}R_k\right)\right) \tag{7}$$

采用式(7)是为了融合2种互补的奖励信号(R_{sys} 和 R_{reg})的优势，并确保奖励信号在0～10合理范围内。R_{sys} 确保学习最终指向任务目标(规避WAF)，R_{reg} 提供即时的、指导性的过程反馈。求和操作使智能体同时兼顾短期(单步优化效果)和长期(最终目标)收益。

2.3.4 交互数据利用

交互数据利用是本框架的一个可选的部分，RL探索中，智能体与环境(即真实WAF及符号增强网络)交互生成的轨迹形式为：〈原始载荷：动作0，奖励0；变异后载荷：动作1，奖励1；…〉。若某次变异后的载荷仍被WAF判定为恶意(即未规避)，则该载荷与其标签(malicious)可构成一个带标签样本对。这些中间结果实质上构成了对WAF决策边界的一次次“探查”，隐含了丰富的规则逻辑信息。因此，本框架可将这些轨迹数据重新组织为增量式数据集，再次用于WAF规则提取。更新后的规则再次转换为更高精度的符号增强网络，进而作为更准确的奖励模型反馈至强化学习训练过程中，以上过程可以继续进行往复。

3 实验论证

为验证框架的有效性，本文进行了系统的实验评估，通过设置端到端多种子组进行5次独立实验并报告平均值。

3.1 实验环境

3.1.1 软件

编译器：python3.8；

WAF软件：Modsecurity、NGX_lua_WAF和Janusec；

DRL算法：基于stable baseline3实现的DQN，PPO，Random Agent算法。

本文的基线方法为直接使用结果级奖励的黑盒测试方法(即稀疏奖励的环境)，以及使用随机网络蒸馏驱动智能体探索的黑盒测试方法。

3.1.2 指标

1) 攻击成功率(ASR)。不重复(若同一个载荷的不同变种都能规避WAF，则只计入1次)的成功规避载荷数占所有待变形的恶意载荷数的比例。

2) 准确率(Accuracy)。在窃取WAF规则时，以WAF的反馈为真值，模型检测正确的样本数占总样本数的比例。

3) 攻击预算(Budget)。对载荷进行变形的

次数，达到最大值时没有规避即判定为失败。

4）假阴性率（FNR）。漏检样本占所有真实阳性样本数的比例，能够衡量 WAF 的检出能力。

3.1.3　数据集

数据集为 Kaggle 中的 SIK 数据集与自建数据集 MDD，MDD 数据集由 100 个经过挑选的不同攻击类别的 SQL 注入组成。

3.2　WAF 规则提取效果与 RNN 模型等效转换效果验证

本节验证了 WAF 规则提取效果，将 WAF 的反馈作为真值，同时保留原始标签用于后续计算假阴性率。根据 2.1 节的方法重新标记的数据生成了正则表达式，示例如图 5 所示，其中，左侧是原始的 WAF 规则，右侧是推断出的规则。

origin regular expression	generated regular expression
1.@rx(?i:merge.*?using\s*?\(|execute\s*?immediate\s*?[\"'`]|match\s*?[\w(),+-]+\s*?against\s*?\()	.* execute \s+ immediate \s+ .*
2. @rx(?i)union.*?select.*?from	.*union.*select.*from.*
3.@rx(?i:(?:[\"'`](?:;.*?\s*?waitfor\s+(?:delay|time) \s+[\"'`]|;.*?:\s*?goto)	.*[\"'];.*\s+waitfor \s+ delay \s+'.*
4.@rx(?i:(?:select|;)\s+(?:benchmark|sleep|if)\s*?\(\s*?\(?\s*?\w+)	.*select \s+.*benchmark \s+.*

图 5　正则表达式生成示例

Fig. 5　Regular expression generation example

生成的正则表达式与原始表达式虽然在实现细节上存在部分差异，比如量词的选用，但是保留其核心的过滤逻辑，因此对检出能力的影响很小。生成的正则表达式性能表现见表 2 所列，提取的 WAF 规则与实际规则相比，准确率达 85%。同时部分生成的正则表达式是原始规则中的子式，这是因为部分恶意载荷具有不少于 2 个的最小匹配单元，所以会多重触发 WAF 规则。而本文方法只保留 1 个最小匹配单元，因此会遗漏部分过滤规则，这类载荷在后续统计中约占 23%。

在生成正则表达式的基础上，本文继续对其进行循环神经网络的等价转化。该符号增强网络直接由符号规则产生，无需训练，其表现如图 6 和表 2 所示。模型的平均准确率为 81.92%，AUC 平均值为 0.82，同时拥有较高的 F1 值与召回率。这说明提取的模型具有良好的克隆目标规则的能力，同时相较于未转化的正则式，准确率虽有损耗但仍小于 3%。损耗原因主要来自张量分解。

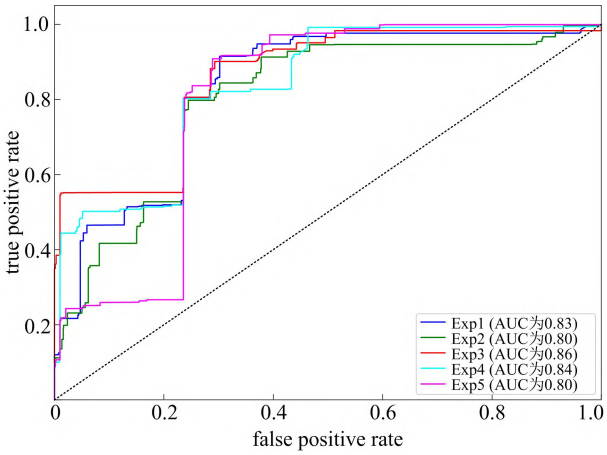


图 6　FA2RNN 的 AUC 曲线

Fig. 6　The AUC curve of FA2RNN

表 2　正则表达式与符号增强网络的性能

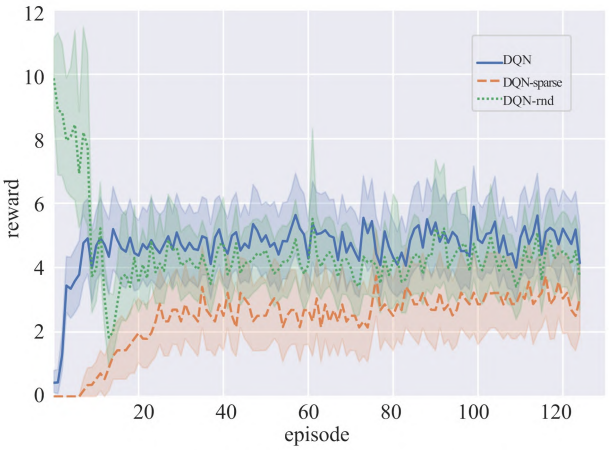
Tab. 2　Regex and symbolic augmented network performance

方法	F1 值	召回率	精确率	准确率
RE	0.841 9±0.010 3	0.838 5±0.023 2	0.842 7±0.037 5	0.848 7±0.006 4
FA2RNN	0.863 3±0.009 1	0.938 2±0.036 4	0.800 9±0.022 1	0.819 2±0.011 7

3.3　基于 RL 的载荷变形性能评估

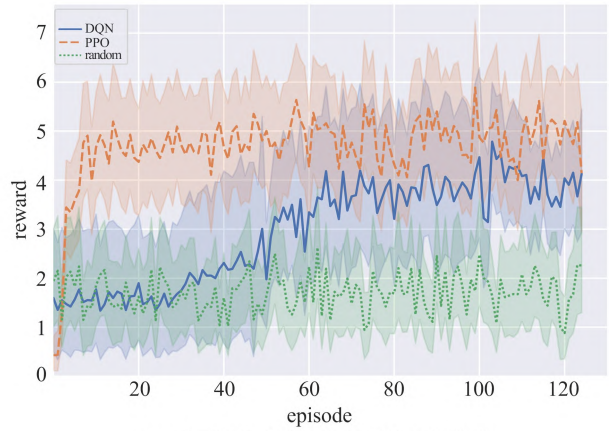
本节以目标 WAF 规则的成功提取为基础展开。通过融合规则提取所得概率模型输出的置信度评分与原始结果级奖励，实现了黑盒攻击向白盒攻击的转换，使 RL 智能体能够获得细粒度的过程级奖励。为验证框架的通用性，使用 2 种经典 RL 算法和随机智能体在框架下进行训练，其平均奖励和回合数的关系如图 7 所示。结果表明，2 种 RL 算法最终均能成功收敛，而随机智能体的平均奖励则始终在较低值区间震荡，这验证了本文所提框架具有一定通用性。同时图 7(c)表明，DQN 算法相比于 PPO 算法收敛更早，且平均奖励普遍高于 PPO 算法，这与 DQN 算法在该环境下对奖励值更加敏感有关。

为验证框架对原本黑盒环境中稀疏奖励问题的缓解效果，使用 2 种经典的 RL 算法，分别在本框架与其他 2 种基线方法下进行训练，平均奖励与回合数的关系如图 7(a)和图 7(b)所示。结果显示，本框架下 2 类算法的平均奖励均高于直接使用结果级奖励的方案。



(a) DQN算法在框架不同方法下的平均奖励

(b) PPO算法在不同方法下的平均奖励



(c) 不同算法在本文框架下的平均奖励

图 7 不同算法在训练时的平均奖励

Fig. 7 The average reward of different algorithms in training

随机网络蒸馏的本质为:智能体借助“目标网络”和“预测器网络”,对环境状态的“新奇度”进行估算并将其作为内在奖励。初期,智能体访问的状态均为全新状态,预测器难以模仿目标网络,由此产生的预测误差(即奖励)较大;随着智能体对相似区域的反复访问,预测器通过学习,对目标网络输出的预测精度显著提高,误差(奖励)随之降低。本框架仅在初期平均奖励低于随机网络蒸馏的方案,但收敛(平均奖励在更短的时间内趋于稳定)速度快于另外 2 种基线方法,这表明本框架能够有效缓解稀疏奖励问题,加速训练收敛进程。

为完整验证框架的功能,即能否有效攻击 WAF 以挖掘其规则漏洞,对不同检出能力的开源 WAF 进行了攻击,同时设置不同的攻击预算以了解攻击预算与 ASR 的关系,结果见表 3 所列。可以看出,在不同检出能力的 WAF 下,该框架均可用于安全测试,且 WAF 漏报率越高,其防护能力越差。MDD 数据集下,由于 ModSecurity_L2 拥有较高的防护性能,因此几乎没有载荷能够规避检测。另外,在攻击预算为 10 的情况下,ASR 均远小于攻击预算为 20 的情况,这说明无限制的情况下,随着攻击预算的增加,ASR 也会随之升高。最后,同样条件下,使用 DRL 对 WAF 进行攻击,成功率基本都高于随机智能体,显示了 RL 在自动化测试的优越性以及其广泛前景。

将本框架与 2 种基线方法的 ASR 进行对比,其攻击预算均为 20,结果见表 4 所列。目标 WAF 为 Modsecurity_L1 时,本框架搭载 PPO 算法的平均 ASR 相较于基线 1 提升了 10.8%,较基线 2 提升了 2.4%;搭载 DQN 算法后较基线 1 提升 9.76%。在面对 Janusec 时,DQN-Ourboros 的 ASR 较基线 1 的 DQN 算法提高了 18.78%,较基线 2 随机网络蒸馏的方法提高了 11.15%,PPO-Ourboros 较基线 1 的 PPO 算法成功率提高了 7.23%,较基线 2 提高了 2.8%。目标防火墙为 Ngx-Lua-Waf 时,DQN-Ourboros 相较于 2 种基线方法分别提升了 13.62%和 4.78%,PPO-Ourboros 分别提升了 6.67%,1.28%。Modsecurity-L2 的规则较为严格,因此规避难度较大,这导致实际规避的载荷数差异小到只有个位数。综上,可以看出缓解稀疏奖励对于本文 ASR 的提升具有一定的帮助,本框架通过设计合成奖励的方式(融合过程级奖励与结果级奖励)来为 RL 提供优化的方向,促使智能体更快找到一种能够规避 WAF 的变形策略,因此在相同预算下攻击成功率更高;但是随着攻击预算的无限增长,ASR 可能会接近,原因在于直接影响 ASR 的主要因素还是 RL 的动作空间,而动作空间决定了智能体的上限,RL 则是在逼近上限。

表 3　攻击预算为 10 和 20 时的攻击结果

Tab. 3　Attack results with budgets of 10 and 20　/%

Dataset	WAF	FNR	Budget=10			Budget=20		
			DQN	PPO	Random	DQN	PPO	Random
SIK	ModSecurity_L1	24.89	51.10	50.35	45.30	59.05	60.24	55.14
	ModSecurity_L2	0.05	0.72	0.69	0.60	1.38	1.41	1.31
	Ngx_Lua_Waf	35.27	72.57	68.25	44.81	76.70	70.20	54.90
	Janusec	51.01	88.21	73.72	70.68	89.20	87.99	80.00
MDD	ModSecurity_L1	1.85	52.73	58.19	30.91	56.36	63.64	51.64
	ModSecurity_L2	0	0	0	0	0	0	0
	Ngx_Lua_Waf	43.64	56.36	58.19	80.36	85.45	89.10	87.27
	Janusec	49.09	85.45	87.27	72.73	87.27	88.73	83.64

表 4　不同算法的实验结果

Tab. 4　Experimental results of different algorithms　/%

Algorithm	ModSecurity-L1	ModSecurity-L2	Ngx-Lua-Waf	Janusec
DQN-sparse	53.80	0.85	67.50	75.10
DQN-rnd	58.50	1.50	73.20	80.25
DQN-Ourboros*	**59.05**	**1.38**	**76.70**	**89.20**
PPO-sparse	54.36	1.00	70.50	82.05
PPO-rnd	58.84	1.20	74.25	85.57
PPO-Ourboros*	**60.24**	**1.41**	**75.20**	**87.95**

4　结束语

本文提出融合 WAF 规则提取与 DRL 的自动化 WAF 测试框架。改进的遗传算法在数据稀缺条件下生成正则表达式，并将其映射为功能上等价的 RNN，精准复现目标 WAF 防护规则。通过设计融合置信度评分与拦截结果的复合型奖励机制，有效缓解了黑盒场景的稀疏奖励难题。

当前变异载荷的手动验证需为每个变体独立部署数据库/后端，验证原始载荷衍生的海量变异体是否保留语义时，工作量将呈指数级增长，远超初始数据集规模。未来，应致力于开发自动化验证系统以替代。

参 考 文 献

[1] APPELT D, NGUYEN C D, BRIAND L. Behind an application firewall, are we safe from SQL injection attacks[C]//Proceedings of the 8th IEEE International Conference on Software Testing, Verification and Validation. [S. l. : s. n.], 2015: 1-10.

[2] ZOU Y H, BAI J J, ZHOU J, et al. TCP-Fuzz: detecting memory and semantic bugs in TCP stacks with fuzzing[C]//Proceedings of 2021 USENIX Annual Technical Conference. [S. l.]: USENIX Association, 2021: 489-502.

[3] WANG Q, CHEN J J, JIANG Z Y, et al. Break the wall from bottom: automated discovery of protocol-level evasion vulnerabilities in Web application firewalls [C]//Proceedings of 2024 IEEE Symposium on Security and Privacy. [S. l.]: IEEE, 2024: 185-202.

[4] ZHENG L K, LI X, WANG C H. ReqsMiner: automated discovery of CDN forwarding request inconsistencies and DoS attacks with grammar-based fuzzing[C]//Proceedings of the 31st Annual Network and Distributed System Security Symposium. [S. l. : s. n.], 2024: 1-18.

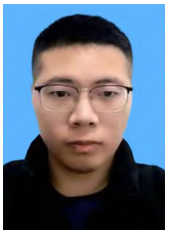
[5] APPELT D, NGUYEN C D, PANICHELLA A, et al. A machine-learning-driven evolutionary approach for testing Web application firewalls[J]. IEEE Transactions on Reliability, 2018, 67(3): 733-757.

[6] YAO Y, HE J J, LI T, et al. An automatic XSS attack vector generation method based on the improved dueling DDQN algorithm[J]. IEEE Transactions on Dependable and Secure Computing, 2024, 21(4): 2852-2868.

[7] LEUNG D, TSAI O, HASHEMI K, et al. XploitSQL: advancing adversarial SQL injection attack generation with language models and reinforcement learning[C]//Proceedings of the 33rd ACM International Conference on Information and Knowledge Management.

New York: ACM, 2024: 4653-4660.
[8] YAN H N, LI X G, ZHANG W J, et al. Automatic evasion of machine learning-based network intrusion detection systems[J]. IEEE Transactions on Dependable and Secure Computing, 2024, 21(1): 153-167.
[9] ISSAKHANI M, HUANG M F, TAYEBI M A, et al. An evolutionary algorithm for adversarial SQL injection attack generation[C]//Proceedings of 2023 IEEE International Conference on Intelligence and Security Informatics. [S. l.]: IEEE, 2023: 1-6.
[10] AMOUEI M, REZVANI M, FATEH M. RAT: reinforcement-learning-driven and adaptive testing for vulnerability discovery in Web application firewalls[J]. IEEE Transactions on Dependable and Secure Computing, 2022, 19(5): 3371-3386.
[11] VALENZA A, DEMETRIO L, COSTA G, et al. WAF-A-MoLE: an adversarial tool for assessing ML-based WAFs[J]. SoftwareX, 2020, 11: 100367.
[12] QU Z Q, LING X, WANG T, et al. AdvSQLi: generating adversarial SQL injections against real-world WAF-as-a-service[J]. IEEE Transactions on Information Forensics and Security, 2024, 19: 2623-2638.
[13] HEMMATI M, HADAVI M A. Using deep reinforcement learning to evade Web application firewalls[C]// Proceedings of the 18th International ISC Conference on Information Security and Cryptology. [S. l.]: IEEE, 2021: 35-41.
[14] CHOWDHARY A, JHA K, ZHAO M. Generative adversarial network(GAN)-based autonomous penetration testing for web applications[J]. Sensors, 2023, 23(18): 8014.
[15] LIANG H L, LI X Y, XIAO D, et al. Generative pre-trained transformer-based reinforcement learning for testing Web application firewalls[J]. IEEE Transactions on Dependable and Secure Computing, 2024, 21(1): 309-324.
[16] THOMPSON K. Programming techniques: regular expression search algorithm[J]. Communications of the ACM, 1968, 11(6): 419-422.
[17] RABIN M O, SCOTT D. Finite automata and their decision problems[J]. IBM Journal of Research and Development, 1959, 3(2): 114-125.
[18] GRIES D. Describing an algorithm by Hopcroft[J]. Acta Informatica, 1973, 2(2): 97-109.
[19] GILES C L, OMLIN C W, THORNBER K K. Equivalence in knowledge representation: automata, recurrent neural networks, and dynamical fuzzy systems[J]. Proceedings of the IEEE, 2002, 87(9): 1623-1640.
[20] AYACHE S, EYRAUD R, GOUDIAN N. Explaining black boxes on sequential data using weighted automata[C]//Proceedings of the 14th International Conference on Grammatical Inference. [S. l.]: PMLR, 2018: 81-103.
[21] JIANG C Y, ZHAO Y G, CHU S B, et al. Cold-start and interpretability: turning regular expressions into trainable recurrent neural networks[C]//Proceedings of 2020 Conference on Empirical Methods in Natural Langulage Procession. [S. l.]: ACL, 2020: 3193-3207.
[22] CYBENKO G. Approximation by superpositions of a sigmoidal function[J]. Mathematics of Control, Signals and Systems, 1989, 2(4): 303-314.

作 者 简 介



陆鹏程
男,1996 年生,硕士研究生,研究方向为人工智能赋能网络安全
E-mail:lupc@nudt.edu.cn

钟晓峰
男,1981 年生,博士,高级工程师,研究方向为网络安全与人工智能
E-mail:zhongxiaofeng17@nudt.edu.cn

陈 杰
男,1992 年生,博士 ,研究方向为多智能体任务分配、演化博弈论以及复杂网络系统的博弈优化
E-mail:jchen202209@163.com

许文博
男,2000 年生,硕士研究生,研究方向为智能渗透测试
E-mail:xuwenbo19@nudt.edu.cn

王永杰
男,1971 年生,博士,教授,研究方向为网络空间安全、智能渗透测试与自动化移动目标防御
E-mail:wangyongjie17@nudt.edu.cn

责任编辑 钱 静