

一种车联网环境下的多阶段恶意网站快速检测方法

毛煜阳¹, 徐崇钧², 杨华雨², 翟溪林¹, 张华¹

(1. 北京邮电大学网络空间安全学院, 北京 100876; 2. 国家计算机网络应急技术处理协调中心浙江分中心, 浙江 杭州 310056)

摘要: 车载第三方服务访问恶意网站的数量迅速增加, 已成为车联网(IoV)安全的重大威胁。目前, 车联网恶意网站检测面临三大瓶颈: 传统工具处理大规模网站数据时存在较长的检测时延, 恶意网站统一资源定位符(URL)混淆问题影响识别准确性, 且恶意网站数据集的获取困难, 这些因素均严重制约了检测的效率和准确性。针对这些问题, 本文提出一种基于逻辑回归的多阶段恶意网站快速检测方法。使用搜索引擎对合法网站进行初步过滤, 减少计算资源的浪费。通过对恶意混淆技术的分析归纳, 设计匹配规则, 并提出一种基于启发式规则的恶意网站过滤方法, 实现混淆网站 URL 的有效过滤, 克服了传统工具无法有效检测带有恶意混淆 URL 的问题。为进一步提升检测准确性, 构建全面且轻量的恶意网站检测特征集合, 并使用逻辑回归分类方法对特征进行提取分析。实验结果表明, 该方法在恶意网站检测的准确性和效率方面显著优于传统方法, 在公开数据集上达到 98.1% 的准确率, 检测时间减少了 75% 左右。

关键词: 车联网; 恶意网站; 过滤与检测; 启发式规则; 机器学习

中图分类号: TP18

文献标志码: A

DOI: 10.19678/j.issn.1000-3428.0070466

A Fast Multi-Stage Malicious Website Detection Method in IoV Environment

MAO Yuyang¹, XU Chongjun², YANG Huayu², ZHAI Xilin¹, ZHANG Hua¹

(1. School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China;

2. Zhejiang Branch of National Computer Network Emergency Response

Technical Team/Coordination Centre of China, Hangzhou 310056, Zhejiang, China)

【Abstract】 The number of malicious websites accessed by in-vehicle third-party services has been rapidly increasing, posing a significant threat to the security of the Internet of Vehicles (IoV). Currently, three major challenges exist for IoV malicious website detection: traditional tools exhibit high detection latency when processing large-scale website data, the presence of obfuscated malicious Uniform Resource Locators (URLs) reduces detection accuracy, and the difficulty in obtaining malicious website datasets further hinders effective detection. Collectively, these factors limit the efficiency and accuracy of detection. To address these issues, this study proposes a fast multi-stage malicious website detection method based on logistic regression. This method uses search engines for the preliminary filtering of legitimate websites to reduce computational resource wastage. It designs matching rules through the analysis and summarization of malicious obfuscation techniques and introduces a heuristic rule-based malicious website filtering method to effectively filter obfuscated website URLs, overcoming the limitations of traditional tools in detecting URLs with malicious obfuscation. To further enhance the detection accuracy, it constructs a comprehensive and lightweight set of malicious website detection features and employs logistic regression classification for feature extraction and analysis. Experimental results demonstrate that the proposed method significantly outperforms traditional methods in terms of accuracy and efficiency in malicious website detection, achieving an accuracy of 98.1% on public datasets and reducing the detection time by approximately 75%.

【Key words】 Internet of Vehicles (IoV); malicious website; filtering and detection; heuristic rules; machine learning

0 引言

随着第五代移动通信技术(5G)的广泛部署, 车联网(IoV)的覆盖范围和数据传输能力得到了显著提高。其中, 智慧车载系统搭载了大量的第三方服

务, 不仅提高了人们的出行体验, 而且显著提升了交通系统的效率^[1]。然而, 互联性的增强也导致车辆面临更加严峻的网络安全挑战^[2], 车辆在连接互联网时会通过车载浏览器链接到恶意网站。恶意网站产生的威胁不仅局限于车辆本身, 还可能对驾驶员

基金项目: 国家自然科学基金(62472047); 浙江省重点研发计划(2024C01012)。

作者简介: 毛煜阳, 男, 硕士研究生, 主研方向为网络与信息安全; 徐崇钧(通信作者), 助理工程师; 杨华雨, 高级工程师; 翟溪林, 硕士研究生; 张华, 教授。

收稿日期: 2024-10-10

修回日期: 2025-02-14

E-mail: 973153326@zju.edu.cn

和乘客的隐私安全、数据完整性,乃至公共安全构成重大威胁^[3]。国家计算机网络应急技术处理协调中心(CNCERT)指出^[4],恶意网站可利用劫持车载网络连接或公共 Wi-Fi 热点等手段实施中间人攻击(MITM),导致用户访问的合法网站被重定向至恶意网站,从而使用户的敏感信息面临被拦截或篡改的风险。

在车联网环境中,车辆间的信息交互以娱乐内容和动态数据的传播为主,但恶意网站可能通过伪造信息、诱导用户访问危险链接等方式对驾驶决策产生误导。因此,恶意网站检测须具备实时性^[5],能够在短时间内完成风险识别以规避潜在威胁。同时,车载设备通常受限于计算能力和存储资源,这对检测算法的复杂度和资源消耗提出了严格约束。与传统环境中依赖高性能服务器或云计算资源的检测方法相比,车联网环境中的检测方案需更加注重算法的轻量化设计与高效性^[6],以适应资源受限的分布式应用场景。因此,在有限的计算资源条件下,实现高效且高精度的恶意网站检测已成为保障车联网环境安全的关键任务。文献[7]利用了朴素贝叶斯等多种机器学习模型进行恶意统一资源定位符(URL)检测,能够达到 96% 的准确率。文献[8]采用了 k 近邻(k -NN)算法,依据相似性进行恶意网站检测,准确率达到 96%,平均检测耗时约为 400 ms。虽然上述研究能够有效地检测恶意网站,但在网站规模较大的情况下进行全量检测会造成计算资源的浪费以及检测时延的加长。

文献[9]通过实验分析恶意 URL 字符分布、字符 n -gram 特征和随机性,揭示了恶意 URL 与合法 URL 之间的显著差异,从而证明恶意网站 URL 存在混淆。文献[10]通过分析恶意 URL 的多种混淆技术(如使用 IP 地址代替域名、拼写错误、URL 缩短服务等),说明恶意网站 URL 存在混淆是为了隐藏其恶意意图,从而绕过传统的黑名单检测和其他安全防护措施。文献[11]在采用决策树的基础上,构建了新的启发式规则库。文献[12]提出了一种基于逻辑回归的方法来识别恶意网站的 URL。上述方法能够通过挖掘恶意 URL 的特征来提高检测准确率,但是无法有效地过滤混淆 URL,并且缺乏恶意词汇、域名服务器(DNS)信息等关键特征的支持,导致检测准确率提高有限。

文献[10]还指出:恶意 URL 的动态性和多样性使得全面收集数据变得困难;标签数据的获取依赖大量人工标注或有限的黑白名单资源;此外,某些特征(如主机信息)的获取受限于时间和成本。这些

因素共同导致恶意网站数据集的获取面临挑战。

为了解决上述问题,本文提出了一种基于逻辑回归的多阶段恶意网站快速检测方法,采用三阶段检测流程,实现了对恶意网站快速有效的检测。本文主要贡献如下:

1)提出一种基于 Google 搜索引擎的合法网站过滤方法,利用浏览器检索特性以及检索效率,通过直接搜索的方式实现对合法网站的快速过滤。

2)提出一种基于启发式规则的恶意网站过滤方法,通过对常用的混淆技术的分析归纳提出 9 条匹配规则,实现对混淆网站 URL 的有效过滤。

3)提出一种基于逻辑回归的检测方法,对无法通过搜索引擎以及启发式规则判断的网站进行多维度的特征收集,构建轻量的恶意网站检测特征集合,通过 L1 型逻辑回归模型对 URL 进行判断,并在公开数据集上进行测试。

本文研究车联网环境下的多阶段恶意网站快速检测问题。引言部分介绍了恶意网站检测的背景、安全挑战及研究现状,提出了优化检测效率与准确性的研究方向。第 1 章总结了 URL 结构、DNS、Whois 信息等概念,并回顾了现有的检测方法。第 2 章提出了一种基于逻辑回归的多阶段恶意网站检测方法。第 3 章通过实验验证了该方法的有效性和时效性。第 4 章总结了本文的主要研究成果。

1 相关知识及相关工作

1.1 相关知识

1)URL 及域名。

URL 是互联网上用于定位和访问资源的地址,通常包括协议、域名、端口(可选)、路径和查询字符串(可选)。域名是 URL 的重要组成部分,用于唯一标识互联网上的计算机。域名可以分为多级,如顶级域名(TLD),包括国家和地区顶级域名(如“.cn”)、通用顶级域名(如“.com”)。

2)DNS 服务器。

DNS 服务器负责将用户输入的域名转换为计算机识别的 IP 地址,实现域名与 IP 地址的映射,便于用户通过域名访问网站。

3)Whois 信息^[13]。

Whois 信息用于查询互联网资源(如域名和 IP 地址)的注册信息,包含注册者的姓名、联系方式、注册日期和到期日期等信息。

4)恶意攻击词。

恶意攻击词是 URL 或域名中诱导用户点击的

关键词,通常与非法活动相关,车联网环境中常见的攻击词包括“car update”“GPS navigation”等。攻击者还可能利用车载系统的信任机制,通过伪造应用或指令植入恶意软件。本文构建了包含 530 个恶意攻击词的词库,用于识别恶意 URL 并初步判断其类型。

1.2 相关工作

目前恶意网站检测技术主要分为 3 类:基于黑白名单、基于启发式规则以及基于机器学习。

基于黑白名单的检测方法是最早应用到恶意网站检测中的方法,如 Google 使用的 Google Safe Browsing^[14]就是该技术的典型应用。文献[15]通过分析可视链接和实际链接之间的相关关系来确定白名单,随后计算域名与已知可信站点的相似度,并作出判断。文献[16]对原有的恶意网站黑名单进行扩展,利用相似度匹配算法对域名与黑名单进行匹配来检测恶意网站。文献[17]采用邻域搜索策略主动发现潜在的未知恶意 URL,增强了黑名单的主动防御能力。但是,攻击者可以使用域名批量生成方法以较低的成本生成并注册大量的虚假域名,增加了黑白名单的实时更新和维护的难度。此外,攻击者可以使用短链接、字符编码等手段混淆恶意 URL,以规避黑名单检测。

启发式检测方法主要基于专家知识或机器学习技术构建规则库,利用规则匹配技术确定网站的性质。文献[18]基于双重支持向量机(SVM)分类器,利用超链接和 URL 的特征检测恶意网站。文献[19]从训练后的 SVM 模型中提取出匹配规则。文献[20]提出了基于关键词和搜索引擎的检测方法 CANTINA,当待检测网站的关键词没有出现在搜索引擎的 Top-N 个搜索结果中,则被判断为恶意网站。文献[21]基于恶意 URL 的特征,构造了 12 条规则进行检测。基于启发式规则的恶意网站检测方

法能够避免复杂的计算,实现对恶意网站的快速判断,满足实时检测的需求,但是存在对规则库规模依赖程度高、更新不及时、维护成本高以及规则之间可能存在冲突等问题。

基于机器学习的方法将恶意网站的检测视为二分类问题,通过从训练样本中提取 URL、网页内容以及主机信息等特征,送入分类器中实现恶意与合法的分类。例如:文献[12]提出了一种基于逻辑回归的方法来识别恶意网站的 URL;文献[22]基于词法特征对恶意网站进行分类。与基于黑白名单及启发式规则的检测方法相比,基于机器学习的检测能够有效识别新出现的恶意网站,并展现出较强的泛化性。然而,恶意网站的攻击者可能会运用 URL 混淆技巧来改变其字符特性,以规避检测工具的识别。此外,机器学习方法在提取 URL 和网页特征,尤其是当特征维度较高时,可能会出现明显的时间延迟。因此,若未进行有效过滤而直接进行全量检测,将导致不必要的计算资源消耗,且难以达到恶意网站实时检测的要求。

2 MSHL 检测方法

为了实现对恶意网站有效、实时的检测,本文提出了基于逻辑回归的多阶段恶意网站快速检测方法 MSHL。MSHL 采用了基于搜索引擎、启发式规则以及逻辑回归 3 种过滤方法,并将其分配到检测的 3 个阶段。通过成熟的搜索引擎中的过滤技术可以快速过滤合法网站,避免对常见合法网站的重复检测,减少检测次数,降低计算资源的消耗;启发式规则能够对采用逃避技术的恶意网站实现准确的检测,并及时更新以应对新的混淆方法;最后综合 DNS 风险度、恶意词汇相似度等多维度特征,使用 L1 型逻辑回归分类器进行快速、准确的判断。检测方法如图 1 所示,分为 3 个阶段:

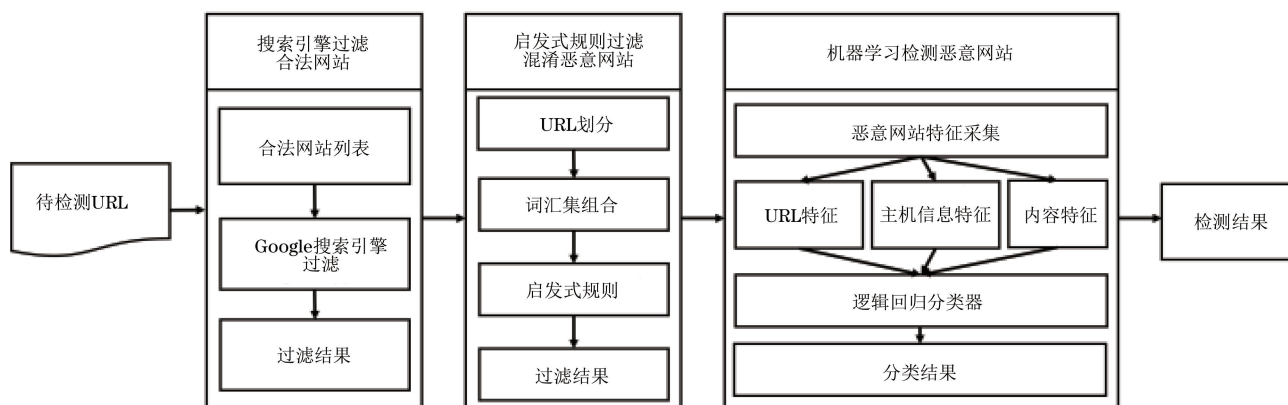


图 1 检测方法整体结构

Fig.1 Overall structure of the detection method

1)搜索引擎过滤。该阶段首先使用白名单来判断当前网站是否为合法网站,然后使用 Google 搜索引擎对待检测网站 URL 或域名进行搜索。若在指定的搜索结果集中存在待检测网站 URL 或域名,则可判定该网站是合法网站。

2)启发式规则过滤。若第一阶段无法直接判定网站为合法网站,进一步通过 URL 划分方式将 URL 划分为不同的词汇集并进行组合,由恶意网站 URL 的字符特征定义的启发式规则对网站进行判定。

3)逻辑回归检测。对于前两个阶段均无法判断的网站,提取该网站的 URL、主机信息以及页面内容等特征,使用逻辑回归分类器判定网站性质。

2.1 基于搜索引擎的合法网站过滤方法

该方法利用搜索引擎内置的 SEG (Search Engine Guide)、PageRank 排名、内容索引等技术快速定位到与查询内容最相关的合法网页。

按照搜索引擎的搜索算法以及排序规则,与搜索目标最相近的结果会被优先展示。由于非法 URL 并不会被收录,如果首页的搜索结果中出现了与待检测 URL 或者域名关键字相同的内容,则证明当前 URL 是合法的。合法网站的 XPath 路径可能因为网站的动态内容和个性化设计而表现出不固定的层级结构。这意味着目标 URL 在网站结构中的具体位置可能会随页面内容的变动而变化。相反,非法网站的链接在搜索引擎中的 XPath 路径往往呈现出固定的模式。合法网站过滤流程为:

1)接收一个 URL 后首先检查其是否在白名单

中,如果当前 URL 在白名单中,将网站标记为合法网站,流程结束;如果当前 URL 不在白名单中,进入步骤 2)。

2)检测当前 URL 是否符合 URL 规范,如果当前 URL 不符合 URL 规范,则进行补充和修正。利用 Google 搜索引擎对 URL 进行搜索,获取前十个搜索结果。

3)分析搜索结果的 XPath 路径,检查固定的 XPath 路径中是否存在 URL 信息,如果路径对应的元素中存在 URL 信息并且与待检测 URL 内容不匹配,则判断网站为未定性网站,结束流程;否则进入步骤 4)。

4)对搜索引擎的前十条搜索结果的 XPath 路径进行匹配,如果匹配到待检测 URL 或者域名,将网站标记为合法网站,结束流程;如果没有匹配到待检测 URL 或者域名,则将网站标记为未定性网站,并将待检测 URL 输入启发式规则匹配过滤方法进一步判断。

2.2 基于启发式规则的恶意网站过滤方法

该方法主要对第一阶段中未定性网站进一步检测。本阶段通过分析目前流行的恶意 URL 混淆方式,设计了 3 种 URL 划分方法,并提出了 9 条用于识别恶意混淆 URL 的规则。

2.2.1 URL 划分方法及恶意混淆 URL 匹配规则

通常恶意 URL 的恶意特征都隐藏在域名的字符串中,因此本节首先给出 3 种划分 URL 的方法,而经划分后的 URL 会产生相应的词汇和子串的集合。3 种 URL 划分方法如表 1 所示。

表 1 URL 划分方法

Table 1 URL segmentation methods

方法编号	划分方法	举例
1	不包括“_”和“-”的任意非字母、非数字符号划分 URL	'http://www.example.com/product/item1'划分为 ['http:', '/', 'www.example.com', 'product', 'item1']
2	以“/”和“\”作为分割点划分 URL	'http://www.example.com/product/item1'划分为 ['http:', '/', 'www.example.com', 'product', 'item1']
3	包括“_”和“-”的任意非字母、非数字符号划分 URL	'http://www.example-data.com/product/item_1'划分为 ['http:', '/', 'www', 'example', 'data.com', 'product', 'item', '1']

合法 URL 中不存在多个标识名与当前网站无关的情况,但是恶意网站往往会在路径中混淆一些合法但是与域名主题不同的字符串,从而绕过以合法字符为特征的检测器,例如“https://malicarnav.com/gps_service/trusted_gps/secure_location/login.php”试图通过路径部分“/gps_service/trusted_gps/secure_location/”来伪装成一个提供 GPS 服务或车联网安全位置服务的合法网站。然

而,真实的 URL 主题标识名(即域名部分)是“malicarnav.com”,这表明它可能是一个恶意网站。为了更好地检测使用了此类混淆手段的恶意网站,本节建立了包含 230 个恶意攻击目标词汇的集合。同时,为了有效检测此类恶意网站,对应提出了规则 1 和规则 2。

规则 1 使用划分方法 1 对目标 URL 进行分割处理后,如果在路径部分检测到恶意关键词,并且

这些关键词与域名中的二级或三级域名不一致,则判定当前网站为恶意网站。

规则 2 使用划分方法 1 对目标 URL 进行分割处理后,如果发现分割后的内容中存在超过一个恶意关键词,则将当前网站判断为恶意网站。

针对采用数字、符号等元素作为连接字符对 URL 进行混淆的情况,本节定义了规则 3~规则 5 来进行检测。其中,规则 4 旨在检测含有分隔符的 URL 混淆情况,在应用过程中,对于网页标识名长度少于 4 个字符的情况,不采用规则 4 进行判断。

规则 3 使用划分方法 1 对目标 URL 进行分割后,即使最初未检测到恶意关键词,但若移除字符串中的数字后识别出恶意关键词,则判定当前网站为恶意网站。

规则 4 在分析目标 URL 时,若初步使用划分方法 1 进行划分后未发现任何恶意关键词,则转而采用更深入的分析策略。该策略包括两个关键步骤:首先,根据划分方法 2 将 URL 分割为多个部分;其次,对每个部分内部再应用划分方法 3 进一步细分,同时在这一过程中排除所有顶级域名相关的词汇。执行完这些操作之后,整合各部分中连续索引位置的字符串片段,如果在其中发现包含恶意关键词或其子序列,则判定该网站为恶意网站。

规则 5 如果域名为显式的点分十进制、十六进制或八进制格式的 IP 地址,如“192. 0. 2. 1”“0xC0. 0x00. 0x02. 0x01”“0300. 0000. 0002. 0001”,即使用了任一格式的 IP 地址作为 URL 的域名,则判定该网站为恶意网站。

为了使恶意网站伪装得更像合法网站,攻击者常在 URL 的非协议部分插入协议字符串。如“http: // mlidomain. com/parm/? https: // www. danger. gps_service. co. cn/login”是一条试图伪装成 GPS 服务的仿冒类恶意 URL。因此,本节提出了规则 6 来识别这类混淆方法。

规则 6 若在 URL 的非协议字段部分(域名、路径等内容)发现协议字段字符串,则判定该网站为恶意网站。

在恶意 URL 构建中,使用顶级域名混淆是一种普遍且有效的逃避手段。攻击者通过在路径或参数部分非传统位置插入众所周知的顶级域名(如“. com”“. net”等),制造了一个外观上合法的 URL,从而误导用户。这种策略是基于用户对于域名结构的一般认知,即域名通常位于 URL 的特定部位。用户往往会忽略域名出现在路径或参数中的异常情况,从而降低了对 URL 真实性的质疑。本节提出

规则 7 识别这类混淆方法。

规则 7 使用划分方法 2 以及划分方法 3 对 URL 进行划分,由于顶级域名的特殊性,如果顶级域名出现在整个域名除末尾以外的位置,则判定该网站为恶意网站。

攻击者通过字符大小写混淆创建与知名网站相似的 URL,以欺骗用户和绕过基于文本匹配的安全措施。例如,将字母“o”替换为数字“0”或者使用特殊符号,可以在不引起怀疑的情况下,改变 URL 的实际目的地。这种细微的替换可能被用户忽略,导致误信和误点击。此外,这种替换通常很难被简单的自动化工具检测到,能有效地规避一些基本的安全检查。因此,本节提出规则 8 来检测这类混淆方法。

规则 8 使用划分方法 2 对 URL 进行划分,解析出 URL 的所有域名部分。在此过程中,需要对每个字符进行检查,以确定是否存在混淆或替换,其中包括比较每个字符及其标准形式,以识别潜在的混淆。例如,数字“0”可能替代字母“o”,或者特殊符号如“\$”可能替代字母“s”。使用 Levenshtein 距离对当前解析得到的域名与合法域名列表的内容进行对比,当计算结果超过 0.7 且小于 1.0,说明当前域名通过改变个别字符仿冒了合法域名,因此判定该 URL 为恶意网址。

关键字堆砌是构造恶意 URL 中常见的技巧,通过在 URL 的路径或查询参数中填充大量相关或不相关的关键字,用于提高其在搜索引擎中的可见性或吸引特定目标受众。在合法网站的设计中,一般会避免使用过量的关键字,以保持 URL 的清晰度和可读性。相反,恶意网站可能故意利用这些关键字来吸引用户,尤其是通过搜索引擎寻找特定内容的用户。因此,本节提出规则 9 来识别这类混淆方法。

规则 9 使用划分方法 1 对 URL 进行划分,统计出现在恶意 URL 关键词列表中的关键词数量,如果一个 URL 的路径或查询参数包含 5 个或更多的关键词,或者这些关键词占整体部分的 50% 以上,则判断该 URL 为恶意网址。

2.2.2 启发式规则过滤恶意网站流程

该阶段接收第一阶段未被判定为合法网站的未知网站,首先将待检测 URL 按照 3 种 URL 划分方式进行划分。将划分结果通过 9 条规则进行匹配,如果匹配到 1 条及以上数量的规则,则说明当前网站使用了规则对应的恶意混淆方式,判定其为恶意网站,若所有规则都未匹配成功,则将当前网站输入第三阶段进行判断。具体流程如图 2 所示。

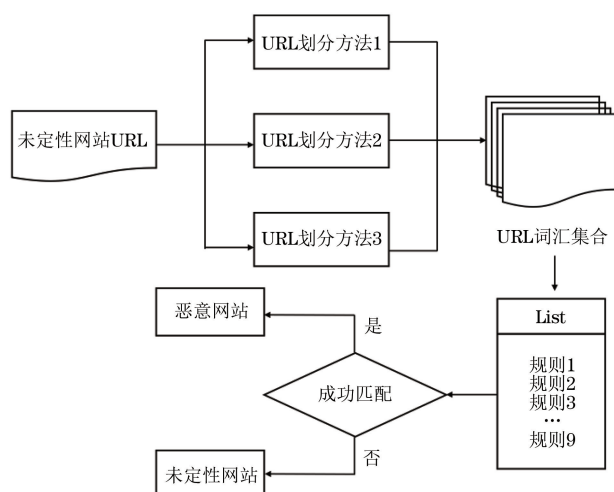


图 2 启发式规则过滤恶意网站流程

Fig.2 Heuristic rules-based process for filtering malicious websites

具体步骤如下：

1)接收搜索引擎过滤阶段中未被判定为合法网站的 URL。

2)按照 3 种 URL 划分方法对未定性网站 URL 进行划分,得到不同的 URL 词汇特征集合。

3)对词汇特征集合的组合使用 9 条启发式规则进行匹配。

4)如果成功匹配到 1 条及以上的规则,则记录匹配的规则和对应的恶意混淆方式,并判定网站为恶意网站;如果没有匹配任何规则,则将当前网站送入第三阶段进行分析与检测。

2.3 基于逻辑回归分类器的检测

本节为恶意网站识别的第三阶段,通过构建恶意网站 URL 以及网页的恶意特征集,使用逻辑回归的方法对恶意网站的性质进行判断。

2.3.1 特征集构建

1)URL 字符特征。

(1)字符信息熵。信息熵主要用来衡量信息复杂度。种子的自动化 URL 生成技术会使恶意网址充斥着大量随机产生的字母与数字字符,这些随机字符序列由于其高度的不确定性表现出较高的信息熵。针对随机生成技术生成的恶意 URL,本节将计算 URL 中字母和数字的信息熵,并将其作为检测特征。信息熵计算公式如式(1)所示:

$$H(S) = - \sum_{i=1}^n p(x_i) \lg p(x_i) \quad (1)$$

式中: $H(S)$ 表示字符串 S 的熵; n 是字符串中不同字符的数量; x_i 是字符串中的第 i 个唯一字符; $p(x_i)$ 是字符 x_i 在字符串中出现的概率,即该字符出现的次数除以字符串的总长度。

(2)字符混淆特征。恶意攻击者常通过混用大小写字母或结合字母和数字的方式来对 URL 进行混淆。因此,本节将这类混淆特征作为恶意网站检测的特征之一。为了有效提取该特征,首先,采取划分方法 3 划分 URL 字符串,得到字符串中的词汇集合。然后,在不考虑首字母大写的前提下,对 URL 内大写字母出现的次数进行统计。最后,统计 URL 中“A-B-A”和“B-A-B”(A 代表数字,B 代表字母)这两种模式的词汇数量。

(3)URL 内容集合特征。本节将待检测 URL 划分为域名、路径内容集合、参数集合以及文件内容集合,并从每个部分中提取包括总长度、最长单词长度、字符串集合长度、“-”或“_”这类非字母和数字符号的数量以及“.”符号的出现次数等特征信息。同时,本节将对 URL 最长词汇长度、URL 域名长度等内容进行特征提取,提取的特征数量为 30 个。

2)Whois 特征。

恶意网站为了避免查封,会频繁更新域名,其域名存活的时间会比合法网站要短,因此本节将提取域名的注册时间、更新时间、失效时间等内容作为恶意网站识别的部分特征。但由于隐私保护策略,通过个人身份注册可以隐藏 Whois 记录中的联系信息等信息来躲避追查和检测。因此,本节将是否以个人身份进行注册、注册邮箱是否匿名等信息作为恶意网站识别的部分特征。考虑到某些注册机构或者所在地区频繁出现恶意攻击行为,本节将 IP 是否被锁定、域名注册机构、IP 是否存在于 IP 自治系统黑名单中等特征作为恶意网站识别的部分特征。Whois 信息的特征共计 8 个。

3)恶意词汇相似度。

恶意 URL 的生成通常依赖于时间戳、用户数据、伪随机数生成器、系统硬件、操作系统信息等多种随机或半随机信息源。利用这些信息源作为随机序列生成的种子,依据特定的规则进行组合,从而以较低的代价高效地产生 URL 字符串。

本节通过三元词组的相似度分析降低误报率,并优化检测效率。具体来说,默认相似度为 0,重点关注待检测 URL 中含有的与恶意关键词一致的三元词组数量。如果它达到目标词汇集合大小的一半,则采用 Jaccard 方法来进行相似度计算;反之则采用默认值作为相似度值。例如,恶意关键词“pornhub”的三元词组为{‘por’, ‘orn’, ‘rnh’, ‘nhu’, ‘hub’}。只有当检测的 URL 至少包含 5 个词中的 3 个时,才会计算其 Jaccard 相似度。此外,如果待检测 URL 的域名标签已在恶意词汇库中,

则将其相似度设为 1,从而减少计算需求,提升检测效率。通过这种方法,可以在保证检测质量的同时,降低时间和空间复杂度,其计算公式如式(2)所示:

$$D_{Js} = \frac{|A \cap B|}{|A \cup B|} \quad (2)$$

式中: $|\cdot|$ 为字符串三元词组集合内元素的数量。网站 URL 与恶意词汇相似度 F_{sim} 的计算方式如下:

$$D_{Js_j} = \frac{1}{n} \sum_{i=1}^n \frac{|A_j \cap B_i|}{|A_j \cup B_i|} \quad (3)$$

$$F_{39} = \frac{1}{k} \sum_{j=1}^k D_{Js_j} \quad (4)$$

式中: D_{Js_j} 代表恶意词汇与 URL 词汇集中第 j 个单词的 Jaccard 相似度; n 为词汇集中的元素总数; k 是采用划分方法 1 分割后 URL 词汇集合的元素数量,该类特征表示为 F_{39} 。

4) DNS 风险度。

DNS 服务商为了保证服务器的安全和本身的信誉,通常会实时地对服务网站进行检测,确保网站不会受到攻击或劫持。而攻击者一般会选择在价格低廉、监管不严格的 DNS 服务器上部署恶意网站。传统检测方法是列举出所有解析过恶意网站的 DNS 服务器,并将这些 DNS 服务器记录在黑名单中。但是由于攻击者技术的提升以及网站数量的不断增长,DNS 服务商无法完全隔绝对恶意网站的解析,因此直接使用黑名单会导致一定的误报。针对此问题,本节提出了 DNS 风险度作为评价一个 DNS 服务器的信誉指标,DNS 风险度越高,意味着在该 DNS 服务器上解析的网站为恶意网站的概率越大。计算方法如式(5)所示:

$$F_{40} = \frac{\sum_{i=1}^n \left(W_i \sum_{j=1}^n P_{ij} \right)}{NQ_{total}} \quad (5)$$

式中: W_i 为第 i 个数据集权重,根据数据集的准确性、更新频率和信誉度确定。经常更新且历史准确率高的数据集具有更高的权重,权重在 1~10 之间分配; P_{ij} 为第 i 个数据集中第 j 个 DNS 服务器的使用频率; Q_{total} 是所有数据集的查询总次数; N 为正则化因子,保证风险度在 0~1 之间。

对于无法解析的 URL,本文将其基准质疑度设定为一个中间值 $F_{base} = 0.5$ 。同时使用式(6)来计算风险度:

$$F_{40} = F_{base} \cdot \frac{\sum_{i=1}^n W_i}{N} \quad (6)$$

为了能够保证风险度的实时性和有效性,权重

W_i 和正则化因子 N 会根据实际情况定期更新。

除了 DNS 风险度外,本节还提取了网站的关联 IP 数量、RETRY 值、TTL 值、EXPIRE 值等共计 6 个特征。

2.3.2 逻辑回归分类器

本节汇总 URL 字符特征、Whois 信息特征、DNS 风险度特征、恶意词汇相似度等共计 46 个特征,采用逻辑回归分类方法对网站的恶意性质进行判断。

URL 字符特征向量表示为:

$$\mathbf{V}_c = \{F_1, F_2, \dots, F_{30}\}$$

URL 的 Whois 信息特征向量表示为:

$$\mathbf{V}_w = \{F_{31}, F_{32}, \dots, F_{38}\}$$

URL 的恶意词汇相似度特征向量表示为:

$$\mathbf{V}_m = \{F_{39}\}$$

URL 的 DNS 风险度特征向量表示为:

$$\mathbf{V}_D = \{F_{40}, F_{41}, \dots, F_{46}\}$$

待检测 URL 完整的特征向量表示为:

$$\mathbf{V}_{url} = \{F_1, F_2, \dots, F_{46}\}$$

完成待检测 URL 特征向量的构造后,采用逻辑回归方法进行恶意性判断,逻辑回归的决策函数如式(7)所示:

$$h_{\theta}(\mathbf{V}_{url}) = \frac{1}{1 + e^{-\theta^T \mathbf{V}_{url}}} \quad (7)$$

式中: θ 为权重向量,反映每种特征在识别恶意网站 URL 中的重要性。

逻辑回归的似然函数如式(8)所示:

$$P(y | \mathbf{V}_{url}; \theta) = h_{\theta}(\mathbf{V}_{url})^y (1 - h_{\theta}(\mathbf{V}_{url}))^{1-y} \quad (8)$$

式中: y 为逻辑回归分类器的输出,当 $y=1$ 时,代表 URL 是恶意的,当 $y=0$ 时,代表 URL 是合法的; v 代表真实标签, $v=1$ 代表正类,也就是恶意样本。

逻辑回归的损失函数如式(9)所示:

$$l(\theta) = \sum_{i=1}^n y^{(i)} \ln h(\mathbf{V}_{url})^{(i)} + (1 - y^{(i)}) \ln (1 - h(\mathbf{V}_{url})^{(i)}) \quad (9)$$

损失函数计算了模型输出与真实标签之间的交叉熵损失,通过梯度下降找到最佳的 θ ,以此提高模型对数据的拟合度。

3 实验设置及实验结果

3.1 实验设置

1) 实验数据集。

本文使用的数据集综合了多个开源社区的资源,包括 PhishTank 恶意网站 URL 数据集、Alexa Top Sites 合法网站 URL 数据集(下文简称 Alexa)、DMOZ 合法网站 URL 数据集,以及从 Google Safe Browsing

的 URLBlacklist 和 Yahoo 获取的合法与恶意 URL 数据集。此外,本节还使用了来自 Mendeley、Kaggle 和 UNB 的恶意 URL 数据集。

2) 实验组。

为了验证本文模型在识别恶意 URL 上的有效性,本文共设置 4 组实验。

实验组 1 此组实验用于第一阶段合法网站过滤效果的验证。首先,对比百度、Bing 以及 Google 3 种搜索引擎的检索效果;其次,比较了 CANTINA^[20] 和 SHLR^[23] 2 种过滤方法

实验组 2 此组实验用于第二阶段基于启发式恶意网站检测效果的验证。选取 3 个实验数据集(包含 1 个合法网站数据集和 2 个恶意网站数据集),通过恶意网站规则库进行实验,统计并分析匹配的 URL 的数量。

实验组 3 此组实验用于第三阶段基于逻辑回归的恶意网站检测效果的验证,主要与当前主要的机器学习方法进行比较,采用一组数据集进行实验。

实验组 4 此组实验用于综合验证模型的检测效果、与其他模型的效果对比以及在不同数据集中的适应能力。首先,通过准确率、召回率、F1 值等指标综合评估模型检测的效果;其次,选择具有代表性的恶意网站检测方法作为对象进行准确率、检测时间等方面的对比。最后,将未在训练中使用的合法网站数据集以及恶意网站数据集进行全排列组合,形成数据集组合,通过模型在不同数据集组合上的检测效果来检测模型的适应能力。

3.2 实验结果与分析

3.2.1 基于搜索引擎的合法网页过滤方法实验结果

1) 搜索引擎过滤合法网站实验。

为了验证不同搜索引擎对合法网站的过滤效

果,本实验选择了 Yahoo、Alexa、URLBlackList 以及 PhishTank 4 个数据集进行实验,其中,Yahoo 和 Alexa 是合法网站数据集,其余两个是恶意网站数据集。选择恶意网站数据集验证不同的搜索引擎在恶意网站数据集中是否会出现误报,从每个数据集中随机抽取 5 000 条数据进行实验。不同搜索引擎的检测结果如表 2 所示。

表 2 不同搜索引擎检测网站结果

Table 2 Detection results for websites across different search engines

单位:条

数据集	百度	Bing	Google
Yahoo	3 060	3 880	4 100
Alexa	3 450	4 290	4 970
URLBlackList	20	60	30
PhishTank	40	150	20

由表 2 分析可得:Google 搜索引擎在合法网站数据集 Yahoo 和 Alexa 上的检测率分别达到了 82%和 99.4%,在恶意网站数据集 URLBlackList 和 PhishTank 上的误报率仅为 0.6%和 0.4%;在百度和 Bing 的检测结果中,百度在 URLBlackList 数据集上的误报率为 0.4%,略低于 Google 搜索引擎的 0.6%,误报率低的原因因为百度会屏蔽部分境外网站,因此搜索失败被判定为恶意网站,误报率会较低。这两个搜索引擎的表现均不如 Google 搜索引擎,因此本文选择 Google 作为合法网站过滤的搜索引擎。

2) 合法网站过滤方法对比实验。

为验证过滤效果,与现有的合法网站过滤方法 TF-IDF^[24]、SHLR^[23] 进行比较。实验随机选取了 Yahoo 数据集和 URLBlackList 数据集中各 1 000 条数据进行实验,实验结果如表 3 所示。

表 3 不同合法网站过滤方法过滤效果

Table 3 Filtering effectiveness of various legitimate website filtering methods

过滤方法	Yahoo 过滤数量/条	URLBlackList 过滤数量/条	耗时/s	CPU 占用/%	内存占用/GB
SHLR	810	12	9.0	35	5
TF-IDF	780	25	10.1	40	6
URLSearch	976	5	3.8	25	4

由表 3 分析可得:本文方法在合法网站的过滤数量上达到了 976 条,较 SHLR 和 TF-IDF 方法分别提高了 20.5%和 25.1%,恶意网站的误报数量为 5 条,仅为 SHLR 和 TF-IDF 的 41%和 20%;在资源占用方面,URLSearch 平均 CPU 占用率为 25%,低于 SHLR 的 35%和 TF-IDF 的 40%,CPU 效率相比 SHLR 提升了 28.6%,相比 TF-IDF 提升了

37.5%,同时,URLSearch 的平均占用内存约为 4 GB,SHLR 和 TF-IDF 的内存占用分别为 5 GB 和 6 GB,因此 URLSearch 分别比 SHLR 和 TF-IDF 节约了 20%和 33.3%的内存占用;在检测时间上,URLSearch 的平均运行时间为 3.8 s,相较同等任务下 SHLR 和 TF-IDF 所需 9.0 s 和 10.1 s 的平均运行时间,分别减少了 57.8%和 62.4%。

3.2.2 基于启发式规则的恶意网站检测实验结果

本节主要验证基于启发式规则的恶意网站检测效果,使用了包含 Yahoo、Alexa 以及 Majestic 的合法网站数据集以及包含 PhishTank、Mendeley、

UNB、URLBlackList 的恶意网站数据集进行实验,每个数据集中各选择 2 000 条数据进行实验,得到被恶意 URL 规则匹配的 URL 的数量,具体实验结果如表 4 所示。

表 4 基于启发式规则匹配的恶意 URL 检测结果

Table 4 Detection results of malicious URLs based on heuristic rule matching

单位:条

规则	PhishTank	Mendeley	UNB	URLBlackList	Alexa	Majestic	Yahoo
规则 1	24	85	81	51	0	15	0
规则 2	32	73	512	43	5	10	0
规则 3	48	94	85	578	11	18	3
规则 4	486	27	13	51	0	5	0
规则 5	71	37	56	6	0	5	0
规则 6	72	109	14	62	21	12	1
规则 7	80	517	101	49	5	10	0
规则 8	95	76	108	87	1	0	0
规则 9	112	2	50	93	1	0	1

由表 4 可知:在规则引擎的错误率仅为 1.55% 的情况下,使用 9 条规则即可检测出 51% 的恶意网站。

3.2.3 基于逻辑回归的恶意网站检测实验结果

1) 不同机器学习模型对比实验。

为了验证 L1 型逻辑回归分类器在恶意网站检测任务上的有效性,本节对比了随机森林、支持向量机以及贝叶斯算法等当前比较流行的机器学习模型。实验使用 PhishTank 数据集中的 5 000 条样本数据,对比实验的检测性能均是通过十折交叉验证法得出。实验结果如表 5 所示。

表 5 不同分类器检测性能对比

Table 5 Comparison of detection performance across

different classifiers

%

分类器	准确率	召回率	F1 值
决策树	93.2	92.9	91.1
贝叶斯	92.2	91.9	91.8
逻辑回归(L1)	98.7	98.9	98.0
逻辑回归(L2)	94.0	94.1	94.1
支持向量机	78.8	73.2	71.1

由表 5 可知:L1 型逻辑回归分类器的准确率达到 98.7%,比决策树高 5.5 百分点,比贝叶斯高 6.5 百分点,比 L2 型逻辑回归分类器高 4.7 百分点,比 SVM 高 20 百分点;L1 型逻辑回归分类器的召回率达到了 98.9%,比决策树高 6 百分点,比贝叶斯高 7 百分点,比 L2 型逻辑回归分类器高 4.8 百分点,比 SVM 高 25.7 百分点;L1 型逻辑回归分类器的 F1 值达到了 98.0%,比决策树高 6.9 百分点,

比贝叶斯高 6.2 百分点,比 L2 型逻辑回归分类器高 3.9 百分点,而与 SVM 相比提升则达到了 26.9 百分点。L1 型逻辑回归分类器在准确率、召回率以及 F1 值这 3 个指标上均表现最优,相较其他分类器而言,更适合恶意网站检测任务。

2) L1 型逻辑回归分类器调参实验。

为了使 L1 型逻辑回归分类器达到最好的分类效果,本节通过交叉验证来评估正则化强度(C 值)与容忍度(Tolerance)的组合对模型准确率的影响。实验结果如表 6 所示。

表 6 不同参数对模型准确率的影响

Table 6 Impact of various parameters on

model accuracy

%

C	Tolerance 为 0.000 1	Tolerance 为 0.001	Tolerance 为 0.01
0.001	85.9	90.7	87.4
0.010	85.8	82.2	88.7
0.100	82.6	95.8	97.9
1.000	81.1	92.9	85.3
10.000	86.4	96.8	72.0
100.000	72.5	70.5	94.1

由表 6 可知:在较低 C 值(例如 0.001 和 0.01)的情况下,模型的准确率并不随着 Tolerance 的增加而增加。在 C=0.001 的条件下,Tolerance 为 0.001 时准确率达到 90.7%,而 Tolerance 为 0.000 1 和 Tolerance 为 0.01 时准确率略有下降,这表明在较强正则化下,模型对容忍度较为敏感。在 C=0.1 的条件下,Tolerance 为 0.01 时准确率达到最高的 97.9%;在 C=10 的条件下,

Tolerance 为 0.001 时准确率达到 96.8%,但当 Tolerance 提高至 0.01 时,准确率出现了下降,这说明过低的正则化强度可能导致模型过拟合。综合实验结果,最终选定了 $C=0.1$ 和 Tolerance 为 0.01 的参数配置,在这一组参数的组合下,模型展现出了最优的性能。

表 7 不同阶段组合检测结果

Table 7 Combined detection results at different stages

配置	准确率/%	召回率/%	F1 值/%	总耗时/s	内存消耗/MB	CPU 消耗/%
阶段 1	46.0	45.5	45.8	12.5	512	15
阶段 2	43.4	43.1	41.9	11.0	1 024	20
阶段 3	91.1	92.1	93.5	34.0	1 536	29
阶段 1+2	83.2	83.0	83.5	17.8	2 048	22
阶段 1+3	92.0	92.3	93.4	21.1	2 560	24
阶段 2+3	94.0	93.6	93.5	26.0	3 072	25
阶段 1+2+3	98.1	98.2	98.2	22.3	4 096	27

分析表 7 可得:阶段 1 资源消耗最少,耗时最短,但是准确率较差,只有 46%,这是因为阶段 1 采用浏览器搜索引擎仅对合法网站的过滤,因此准确率等指标数值较低;阶段 2 采用启发式规则匹配的方法进行检测,能够对带有混淆的恶意网站进行检测,但是无法判断网站是否为合法网站,因此检测准确率只有 43.4%,阶段 2 采用的字符串分割及匹配的方法,检测速度较快,总耗时最少,4 000 条数据的总耗时为 11 s;阶段 3 采用机器学习的方法通过提取特征进行检测,能达到 91.1%的准确率,但是相应的总耗时为 34 s。

在组合运行的配置中,阶段 1+2 和阶段 1+3 的组合在准确率和召回率上都有所提升,尤其是阶段 1+3,准确率和召回率分别为 92%和 92.3%,尽管阶段 1+3 的资源消耗(内存消耗 2 560 MB,CPU 消耗 24%)高于阶段 1+2,但总耗时较短(21.1 s),这可能是由于阶段 3 在特定条件下能更快地处理信息。阶段 2+3 准确率和召回率分别为 94%和 93.6%,但是由于阶段 2 需要对所有 URL 进行匹配,因此总耗时要高于阶段 1+3。最终,3 个阶段整体运行的配置在检测性能上表现最优,准确率和召回率均在 98%以上,并且相对于阶段 2+3 的配置,总耗时降低到 22.3 s,但 3 个阶段的组合检测会导致资源消耗有所增加。

3.2.5 MSHL 恶意网站检测效果

本节主要验证 MSHL 模型与当前恶意网站检测模型的对比以及在不同数据集中的适应能力。

1) 不同恶意网站检测工具对比实验。

本节使用的数据集是 PhishTank 和 Yahoo 的

3.2.4 多阶段通用性验证实验

为了验证不同阶段对检测效果的影响,本实验选择 Alexa 和 URLBlackList 数据集中各 2 000 条 URL 样本进行实验,其中 Alexa 为合法网站数据集,URLBlackList 为恶意网站数据集,不同阶段的组合的检测结果如表 7 所示。

恶意网站数据集与合法网站数据集的组合,其次是除去上述两个数据集之外的所有合法网站数据集与恶意网站数据集之间的两两组合。同时选取准确率、召回率、F1 值以及单条识别时间 4 个指标作为评估标准与 UEM^[25]、SHLR^[23]以及 Phishnot^[26]恶意网站检测方法进行对比,以验证本节提出的恶意网站检测方法的有效性和优越性。实验结果如表 8 所示。

表 8 对比实验结果

Table 8 Comparative experimental results

检测方法	准确率/%	召回率/%	F1 值/%	单条识别时间/ms
UEM	92.2	92.8	92.2	360
Phishnot	95.1	95.5	95.1	288
SHLR	96.3	96.2	96.5	272
MSHL	98.1	98.3	98.8	70

分析表 8 可得:相较于其他检测方法,MSHL 的快速过滤以及利用具有代表性且维度更低的网站特征,使得检测流程更加高效。在不考虑网络通信延时的条件下,MSHL 识别单个 URL 的平均耗时仅为 70 ms,相比 SHLR 检测时间缩短约 75%,能够满足实时检测需求,且检测准确率均超越其他 3 种方法,高达 98.1%。综合考虑单个 URL 判定的时间延迟和检测效率,MSHL 优于其他方法。

2) 模型泛化能力实验。

为了验证 MSHL 模型在不同数据集中的适应能力,在与训练数据集不同的数据集上进行验证。本节使用了 DMOZ+URLB、DMOZ+mendeley、DMOZ+UNB、Majestic+URLB、Majestic+mendeley、Majestic+UNB、Alexa+URLB、Alexa+mendeley 以及 Alexa+UNB 共计 9 个数据集进行实验,这些数

数据集均为合法 URL+恶意 URL 数据集的组合。实验结果如表 9 所示。

表 9 不同数据集组合下 MSHL 检测效果

Table 9 Detection effectiveness of MSHL under different dataset combinations

数据集组合	准确率	召回率	F1 值
DMOZ+URLB	94.8	94.0	94.4
DMOZ+mendeley	93.2	92.5	92.8
DMOZ+UNB	95.6	94.9	95.2
Majestic+URLB	94.0	93.5	93.7
Majestic+mendeley	92.5	91.7	92.1
Majestic+UNB	96.1	95.5	95.8
Alexa+URLB	95.1	94.3	94.7
Alexa+mendeley	93.9	93.1	93.5
Alexa+UNB	96.4	95.8	96.1

由表 9 分析可得:在准确率方面,Alexa+UNB 组合最高,达到了 96.4%,Majestic+mendeley 组合的准确率为 92.5%;在召回率方面,Alexa+UNB 组合为 95.8%,效果最优,模型在该数据组合下能够较为全面地识别出恶意 URL;在 F1 值方面,Alexa+UNB 达到了 96.1%,相较之下,Majestic+mendeley 组合的 F1 值为 92.1%。通过实验能够得出 MSHL 模型在不同的数据集组合中的识别准确率以及 F1 值均超过了 91.0%,表明 MSHL 模型在不同数据集上具有良好的适应性。

4 结束语

针对车联网中恶意网站产生的安全威胁,本文提出了一种基于逻辑回归的多阶段恶意网站快速检测方法 MSHL。将恶意网站检测的工作分为 3 个阶段,在不同阶段分别采用不同的检测技术,进一步提升了检测时效性以及准确率。同时,针对被混淆的恶意网站 URL 构建对应的规则库以及更加全面和轻量的恶意网站检测特征集合。通过在 Alexa、PhishTank 等公开数据集上的实验证明,MSHL 在恶意网站检测任务上能够达到 98.1%的准确率,同时,检测时间由 280 ms 减少到 70 ms,检测时间减少了 75%左右。

参考文献

- [1] 孙德强,张俊仪,时瑞浩.车载信息娱乐系统发展及趋势研究[J].汽车电器,2024(6):39-41.
- [2] 刘大鹏.智能网联汽车信息安全挑战与思考[J].中国信息安全,2024(2):41-43.
- [3] 丁志海.车联网网络安全风险态势及对策分析[J].时代汽车,2023(17):194-196.
- [4] 国家计算机网络应急技术处理协调中心.2020 年中国互联网络网络安全报告[EB/OL].[2023-11-10].https://www.cert.org.cn/publish/main/upload/File/2020%20Annual%20Report.pdf.
- [5] 赵占永.基于动静态结合的恶意 URL 分析研判技术[J].广播电视网络,2022,29(10):71-74.
- [6] 江恺,曹越,周欢,等.车联网边缘智能:概念、架构、问题、实施和展望[J].物联网学报,2023,7(1):37-48.
- [7] ALJABRI M, ALHAIDARI F, MOHAMMAD R M A, et al. An assessment of lexical, network, and content-based features for detecting malicious URLs using machine learning and deep learning models[J]. Computational Intelligence and Neuroscience, 2022, 2022: 3241216.
- [8] MEHNDIRATTA M, JAIN N, MALHOTRA A, et al. Malicious URL: analysis and detection using machine learning [C] // Proceedings of the 10th International Conference on Computing for Sustainable Global Development (INDIACom). New Delhi, India: IEEE Press, 2023: 1461-1465.
- [9] VERMA R, DAS A. What's in a URL: fast feature extraction and malicious URL detection[C] // Proceedings of the 3rd ACM International Workshop on Security and Privacy Analytics. New York, USA: ACM Press, 2017: 55-63.
- [10] SAHOO D, LIU C H, HOI S C H. Malicious URL detection using machine learning: a survey[EB/OL].[2023-11-10].https://arxiv.org/abs/1701.07179.
- [11] KHALIFE J, HUSSAIN M NASSAR F T, MARRI M H M K A. New heuristics method for malicious URLs detection using machine learning [C] // Proceedings of the International Symposium on Networks, Computers and Communications (ISNCC). Doha, Qatar: IEEE Press, 2023: 1-6.
- [12] CHIRAMDASU R, SRIVASTAVA G, BHATTACHARYA S, et al. Malicious URL detection using logistic regression [C] // Proceedings of the IEEE International Conference on Omni-Layer Intelligent Systems (COINS). Barcelona, Spain: IEEE Press, 2021: 1-6.
- [13] 薛鹏飞,沈毅,胡森,等.基于规则的域名 WHOIS 信息抽取技术研究[J].信息对抗技术,2023,2(1):66-77.
- [14] Google. Google Safe Browsing [EB/OL].[2023-11-10].https://developers.google.cn/safe-browsing/2020-10-15.
- [15] AZEEZ N A, MISRA S, MARGARET I A, et al. Adopting automated whitelist approach for detecting phishing attacks [J]. Computers & Security, 2021, 108: 102328.
- [16] PRAKASH P, KUMAR M, KOMPILLA R R, et al. PhishNet: predictive blacklisting to detect phishing attacks[C] //

- Proceedings the 29th IEEE Conference on Computer Communications (INFOCOM 2010). San Diego, USA: IEEE Press, 2010: 1-5.
- [17] AKIYAMA M, YAGI T, ITOH M. Searching structural neighborhood of malicious URLs to improve blacklisting[C]// Proceedings of the IEEE/IPSJ International Symposium on Applications and the Internet. Munich, Germany: IEEE Press, 2011: 1-10.
- [18] RAO R S, PAIS A R, ANAND P. A heuristic technique to detect phishing websites using TWSVM classifier[J]. Neural Computing and Applications, 2021, 33(11): 5733-5752.
- [19] MOGHIMI M, VARJANI A Y. New rule-based phishing detection method[J]. Expert Systems with Applications, 2016, 53: 231-242.
- [20] ZHANG Y, HONG J I, CRANOR L F. CANTINA: a content-based approach to detecting phishing web sites[C]// Proceedings of the 16th International Conference on World Wide Web. New York, USA: ACM Press, 2007: 639-648.
- [21] SILVA C M R, FEITOSA E L, GARCIA V C. Heuristic-based strategy for phishing prediction: a survey of URL-based approach [J]. Computers & Security, 2020, 88: 101613.
- [22] KUMARI K, JAISON F. Detection of URL based phishing websites using machine learning in Django framework[J]. International Journal for Research in Applied Science and Engineering Technology, 2022, 10(3): 1151-1153.
- [23] NARU P, REDDY C S K, SEKHAR P G, et al. Detection of fake websites using machine learning techniques [C]// Proceedings of the 3rd International Conference on Smart Data Intelligence (ICSMDI). Trichy, India: IEEE Press, 2023: 477-482.
- [24] ZHANG Y X, FU X Y, YANG R, et al. DRSDetector: detecting gambling websites by multi-level feature fusion[C]// Proceedings of the IEEE Symposium on Computers and Communications (ISCC). Gammarth, Tunisia: IEEE Press, 2023: 1441-1447.
- [25] YAN X D, XU Y, CUI B J, et al. Learning URL embedding for malicious website detection[J]. IEEE Transactions on Industrial Informatics, 2020, 16(10): 6673-6681.
- [26] ALANI M M, TAWFIK H. PhishNot: a cloud-based machine-learning approach to phishing URL detection [J]. Computer Networks, 2022, 218: 109407.

文字编辑 金胡考
栏目编辑 赖玉玲