恶意网页识别研究综述

沙泓州[1),2),3)] 刘庆云[1),3)] 柳厅文[1),3)] 周 舟[1),3)] 郭 莉[1),3)] 方滨兴[2),3)]

[1)](中国科学院信息工程研究所 北京 100093)

[2)](北京邮电大学计算机学院 北京 100876)

[3)](信息内容安全技术国家工程实验室 北京 100093)

摘 要 近年来,随着互联网的迅速发展以及网络业务的不断增长,恶意网页给人们的个人隐私和财产安全造成的威胁日趋严重.恶意网页识别技术作为抵御网络攻击的核心安全技术,可以帮助人们有效避免恶意网页引起的安全威胁,确保网络安全.文中从理论分析和方法设计两方面介绍了恶意网页识别的最新研究成果.在理论分析层面,从恶意网页的基本概念和形式化定义出发,对恶意网页识别的应用场景、基本框架及评价方法进行全面的归纳,并总结了恶意网页识别的理论依据及性能评价指标.在方法设计层面,对具有影响力的恶意网页识别方法进行了介绍和归类,对不同类别的识别方法进行了定性分析和横向比较.在总结恶意网页识别研究现状的基础上,从客观环境的变化以及逃逸技术的升级两方面深入探讨了当前恶意网页识别面临的技术挑战.最后总结并展望了恶意网页识别的未来发展方向.

关键词 恶意网页识别;网页分类;机器学习;逃逸技术

中图法分类号 TP393 **DOI号** 10.11897/SP.J.1016.2016.00529

Survey on Malicious Webpage Detection Research

SHA Hong-Zhou[1),2),3)] LIU Qing-Yun[1),3)] LIU Ting-Wen[1),3)]

ZHOU Zhou[1),3)] GUO Li[1),3)] FANG Bin-Xing[2),3)]

[1)](*Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093)

[2)](*School of Computer Science, Beijing University of Posts and Telecommunications, Beijing* 100876)

[3)](*National Engineering Laboratory for Information Security Technology, Beijing* 100093)

Abstract In recent years, with the rapid development of Internet and the increasing growth of network services and security needs, the existence of malicious web pages have become a much more serious problem for personal privacy and property safety. As one of the key technologies to resist network attacks, the detection techniques for malicious web pages can effectively help people avoid potential security threats and thus ensure the network security. In this paper, we describe the latest research achievements from theory to practice. It starts from the introduction of the formal definition of malicious web pages, and followed by concluding the detection techniques' application scenarios, basic framework and evaluation principles. Then, it introduces several typical detection schemes, classifies them into categories, and finally puts them to a horizontal comparison. Based on the understanding of the research status in malicious web page detection schemes, this paper presents an in-depth discussion of the current challenges in which

收稿日期:2015-01-07;最终修改稿收到日期:2015-05-21.本课题得到中国科学院战略性先导科技专项(XDA06030200)、国家科技支撑计划(2012BAH46B02)和国家自然科学基金项目(61402474)资助.沙泓州,男,1988年生,博士研究生,中国计算机学会(CCF)学生会员,主要研究方向为信息安全、网络安全.E-mail:shahongzhou@foxmail.com.刘庆云(通信作者),男,1980年生,博士,高级工程师,主要研究方向为信息安全、网络安全.E-mail:liuqingyun@iie.ac.cn.柳厅文,男,1986年生,博士,助理研究员,主要研究方向为大数据安全分析、算法分析与设计.周 舟,男,1983年生,博士,高级工程师,主要研究方向为网络安全、高性能网络.郭 莉,女,1969年生,高级工程师,主要研究方向为信息安全、网络安全、数据流处理.方滨兴,男,1960年生,博士,教授,博士生导师,中国工程院院士,主要研究领域为网络安全、信息内容安全.

people have to face, including both dynamical changes of the objective environments and upgrades of the escape techniques. Finally, it looks into the future of this field.

Keywords malicious web page detection; web page classification; machine learning; escape technology

1 引 言

近年来,互联网的蓬勃发展为人们的日常生活创造了巨大的便利条件.与此同时,便捷的网络服务也吸引了网络攻击者们通过钓鱼网站[1]、垃圾广告[2]和恶意软件[3]推广等方式非法牟利.尽管这些不法活动的目的和手段各不相同,但它们都需要不知情的用户访问攻击者提供的网页地址以达到攻击目的.这些网页因此被称为恶意网页.卡巴斯基报告①指出,2012 年,恶意网页在 87.36%的网络攻击中出现,并已成为黑客谋求不法经济利益的重要工具.此外,Google 的研究[4]指出,其搜索结果中 1.3%的页面为挂马网页.因此,如何有效地识别恶意网页已经成为亟待解决的网络安全问题之一.

针对恶意网页识别问题,研究人员提出很多识别技术和解决方案.Ma 等人[5-7]依据 DNS 信息、WHOIS 信息以及 URL 的语法特征,采用机器学习算法对恶意 URL 进行识别.Canali 等人[8]在此基础上增加了 Javascript 和 HTML 特征,从而能够对网页内容进行检测,提升了恶意网页识别准确率.此外,Thomas 等人[9]应用蜜罐技术分析浏览器响应动作(例如 Javascript 事件及弹出新窗口等),以此判定当前访问页面是否为恶意网页.而一些浏览器出于快速响应需要采用内置恶意网址列表的方法为用户提供轻量级的实时恶意网页识别服务(例如 Internet Explorer 浏览器的 SmartScreen 筛选器②及谷歌浏览器的 Safe Browsing③等).随着网络服务日益普及,围绕恶意网页的攻防博弈也在持续进行,攻击者不断采用一些新的技术手段(例如自动生成域名[10]及网页隐匿技术[11]等)来增强恶意网页的隐蔽性,提高攻击效率;而防御方安全研究人员深入研究恶意网页识别技术,不断提出相应的检测手段和防范措施.

从公开发表的科研论文和资料来看,国内外对恶意网页识别的相关研究还不够全面和深入.2012 年,张慧琳等人[12]从网页木马的机理出发,从挂马检测、特征分析、防范技术等方面对网页木马的研究进行了分析和总结,但缺少对各类检测方法的横向比较和讨论;2013 年,诸葛建伟等人[13]对识别恶意网页常用的蜜罐技术的研究发展和应用情况进行了综述,但没有涉及其它恶意网页识别技术.此外,2013 年,Mahmoud 等人[1]对钓鱼网站的离线识别方法进行了详细地分析和总结,但缺少对其他恶意网页以及在线识别方法的分析.

因此,本文尝试对恶意网页识别的思路、方法、技术进行全面的归纳和总结,介绍恶意网页识别系统的整体框架和应用场景,详细分析恶意网页识别的研究现状并探讨防御方目前面临的挑战,为进一步研究作参考.

本文第 2 节介绍恶意网页的基本概念与形式化定义;第 3 节对恶意网页识别技术进行概述;第 4 节和第 5 节对恶意网页识别的研究进展和挑战进行总结和分析;第 6 节对未来工作进行展望并对全文工作进行总结.

2 恶意网页基本概念与形式化定义

2.1 恶意网页基本概念

目前,学术界对恶意网页尚无一个明确的、统一的定义.Google③将恶意网页限定为一种不安全的网站,发生的场景可以是恶意软件自动下载[14],网页弹窗[1]诱骗用户输入自己的用户名和密码等.而 Eshete 等人[15]将恶意网页定义为一类利用漏洞对一次性的访问行为发起攻击的网页.此外,百度百科上④将恶意网站定义为故意在计算机系统上执行恶意任务的病毒、蠕虫和特洛伊木马的非法网站,并指出它们的共同特征是采用网页形式让人们正常浏览

① Kapaersky. Kapaersky security bulletin. http://www.securelist.com/en/analysis/204792255/Kapersky, 2012.10.10
② Internet Explorer. SmartScreen filter. http://windows.microsoft.com/zh-CN/internet-explorer/use-smartscreen-filter#ie=ie-9, 2014.12.12
③ G. T. Report. Making the Web safer. http://www.google.com/transparencyreport/safebrowsing/?hl=en, 2014.12.30
④ Baidu. Definition of malicious Web sites. http://baike.baidu.com/view/2382119.htm, 2014.12.31

页面内容，同时非法获取电脑里的各种数据. 该定义指出了“访问页面时执行恶意行为”和“非法窃取用户数据”两个关键点.

综上所述，恶意网页是一类以网页木马、钓鱼网站为代表的网页. 不同于正常网页，恶意网页通过伪装成合法网站或在页面中嵌入恶意脚本等方式，在用户访问时对其网络安全构成威胁.

因此，本文将恶意网页定义为以网页形式出现，以访问时窃取用户隐私、安装恶意程序或执行恶意代码等恶意行为为目的的网页集合.

2.2　恶意网页识别形式化定义

恶意网页识别问题的本质是一个二分类问题，可以形式化定义如下：设 W 表示网页样本集合：$W=\{w_1, w_2, \cdots, w_i, \cdots, w_n\}$，其中 n 为网页数量，w_i 为第 i 个网页. C 表示网页类标号集合：$C=\{c_l, c_m\}$，其中 c_l 表示良性网页，c_m 表示恶意网页. 则目标函数为从网页样本集合到类标号集合的映射函数：$\Phi(w_i, c_j): W\times C\rightarrow\{0,1\}$. 其中，$1\leq i\leq n$，$j\in\{l,m\}$. $\Phi(w_i, c_j)$ 是一个二分类函数. 因为恶意网页识别只识别网页 w_i 是否属于恶意网页集合，所以，该目标函数可以简化为 $\Phi(w_i, c_j): W\rightarrow\{0,1\}$.

3　恶意网页识别技术概述

本节依据已有的恶意网页识别系统和相关研究工作，首先分析恶意网页识别的主要应用场景，然后对恶意网页识别系统的基本框架进行归纳，并总结相应的评价指标.

3.1　恶意网页识别的应用场景

恶意网页规模的不断壮大，在给人们的个人隐私和财产安全带来威胁的同时，客观上也为恶意网页识别技术创造了庞大的安全市场和广泛的应用需求. 下面将从恶意网页的攻击形式以及检测位置等方面分析恶意网页识别技术的应用场景.

按照攻击形式来分，恶意网页主要分为钓鱼网页[16]和恶意软件下载[12]两大类. 其中，依据中国互联网协会的定义⑤，恶意软件作为一个集合名词，指代在计算机系统上执行恶意任务[17]的蠕虫、病毒、网页木马、间谍软件等. 由于采用的攻击形式不同，这些恶意网页对访问者构成了不同类型的安全威胁. 一些恶意网页常用于窃取用户的个人隐私信息（例如攻击者常利用钓鱼网页窃取用户的银行帐号及密码信息等），而另一些恶意网页则通过下载和执行恶意程序或脚本（如病毒、木马、蠕虫等），对访问者的计算机系统安全构成威胁.

按照检测位置来分，识别恶意网页的位置可以分为 3 类：服务器端（例如搜索引擎[18]及社交网站[19]等），客户端（例如提供安全服务的浏览器插件②、杀毒软件⑥等）以及网关端（例如安全网关设备[20]等）. 其中，大多数恶意网页识别系统[18-19]在服务器端和客户端进行识别和检测.

3.2　恶意网页识别系统基本框架

为了准确识别数量众多、种类多样的恶意网页，恶意网页识别系统应当同时具备可用性与可扩展性. 由于目前已有的恶意网页识别系统大都面向某一类特定应用，因此系统结构和实现方式存在一些差异. 下面结合恶意网页识别的目标和一般规律，提炼出恶意网页识别系统的基本框架，如图 1 所示，该框架主要分为 3 个部分.

（1）网页采集. 负责对互联网上的网页进行收集、去重和过滤. 其中，按照网页收集方式，一般可分为主动和被动两种. 主动收集，主要是运用网络爬虫技术，从互联网中定向抓取网页集合. 而被动收集，主要是在网关或客户端蜜罐中，对经过的访问流量进行采集. 而流量过滤是根据网页的相关信息（例如网页后缀及网页类型等），对明显不属于恶意网页的部分进行过滤，以提升恶意网页识别系统的运行效率.

（2）特征抽取. 特征抽取是指不同的识别方法从识别不同种类恶意网页的实际需求出发，依据网页自身特点，抽取信息作为识别恶意网页依据的过程. 这些特征包括但不限于 URL 词汇特征、主机信息特征、网页内容特征、URL（DNS）黑名单、链接关系以及跳转关系等. 如图 2 所示，针对不同类别的恶意网页，学者们从不同角度提出了很多识别恶意网页的特征. 常用的识别特征按照其来源的不同可以分为静态特征和动态特征两类.

静态特征主要来源于网页静态信息，其种类繁多，但抽取过程相对简单. 常见的静态特征主要包括主机信息（例如 WHOIS 信息[21]及 DNS 信息[22]等），URL 信息（例如词袋特征[23]及 URL 长度[24]等）和网页内容（例如 HTML Tag 信息[25]、Javascript 代码[26]、网页漏洞信息[27]及链接关系[28]）等. 而动态特征主要来自于网页动态行为，其种类较少，但抽

⑤ Definition of malware. http://tech.sina.com.cn/i/2006-11-22/20311251645.shtml, 2014.12.30

⑥ McAfee: Mcafee site advisor. http://www.siteadvisor.com/, 2014

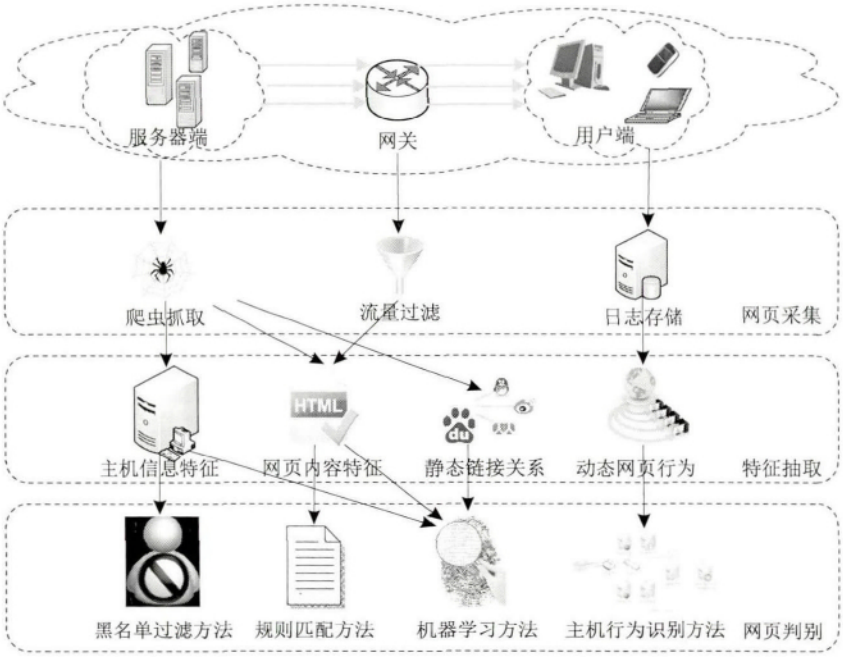

图 1 恶意网页识别的基本框架

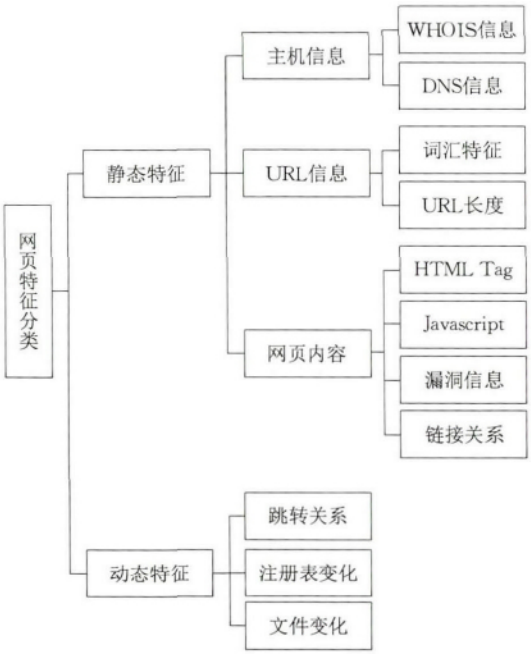

图 2 识别恶意网页的特征分类

取过程相对复杂. 常见的动态特征主要包括浏览器行为[29]、网页跳转关系[30]、注册表及文件夹变化情况[13]等. 这些特征往往需要人们对可疑的网页进行长时间的深入分析才能获得. 同时,在使用动态特征的过程中,往往需要结合蜜网技术⑦和虚拟化技术[31]辅助识别恶意网页. 此外,一些研究从 HTTP 会话[32]、搜索引擎提供的相似网页[33]出发,为识别恶意网页提供新思路.

(3)网页判别. 目前常用的网页判别方法包括黑名单过滤法、规则匹配法、机器学习方法及基于交互式主机行为的识别方法. 本文将在第 4 节对此作详细介绍.

3.3 恶意网页识别评价指标

用于识别恶意网页的评价指标,通常可以分为以下两类:功能指标和性能指标. 其中,功能指标主要用于对恶意网页的识别效果进行评价;而性能指标主要对恶意网页的识别效率进行评价. 为了方便对功能指标进行描述,表 1 展示了常用于衡量分类准确性的混淆矩阵[34]. 其中,$N_{M\to M'}$ 表示恶意网页被正确划分的数量,$N_{L\to M'}$ 表示良性网页被错误划分为恶意网页的数量(常被称为误报),$N_{M\to L'}$ 表示恶意网页被错划分为良性网页的数量(常被称为漏报),$N_{L\to L'}$ 表示良性网页被正确划分的数量. 其中,漏报和误报是恶意网页识别中可能出现的两种错误情况. 举例来说,如有 200 个网页,分为良性网页和恶意网页两类,各 100 个. 表 1 展示了由恶意网页识别系统处理得到的一种可能情况. 在表 1 中,误报 20 个,漏报 10 个.

表 1 分类混淆矩阵

	认定为恶意网页(M′)	认定为良性网页(L′)
恶意网页(M)	$N_{M\to M'}$(如 90)	$N_{M\to L'}$(如 10)
良性网页(L)	$N_{L\to M'}$(如 20)	$N_{L\to L'}$(如 80)

⑦ The Honeynet Project. Capture-HPC. http://projects.honeynet.org/capture-hpc, 2014.10.20

依据混淆矩阵，恶意网页识别的功能指标主要有：

真正类率(True Positive Rate，TPR)．由式(1)计算，表示在所有恶意实例中被检测出的恶意实例的比例．

假正类率(False Positive Rate，FPR)，也称误报率．由式(2)计算，表示在所有良性实例中被错误检测为恶意实例的良性实例的比例．

真负类率(True Negative Rate，TNR)．由式(3)计算，表示在所有良性实例中被正确检测的良性实例的比例．

假负类率(False Negative Rate，FNR)，也称漏报率．由式(4)计算，表示在所有恶意实例中被错误检测为良性实例的恶意实例的比例．

精确度(Precision)．由式(5)计算，表示在所有被检测出的恶意实例中正确的恶意实例的比例．

召回率(Recall)，也称查全率．等价于 TPR，由式(6)计算．

F-measure 是准确率和查全率的加权调和平均，可由式(7)计算得到，其中 β 为参数．当参数 $\beta=1$ 时，就是最常见的 F1-measure，可由式(8)计算得到．

准确度(Accuracy)．由式(9)计算得到，表示在所有实例中正确检测出的良性实例和恶意实例的比例．

$$TP=\frac{N_{M\to M}}{N_{M\to M}+N_{M\to L}} \tag{1}$$

$$FP=\frac{N_{L\to M}}{N_{L\to L}+N_{L\to M}} \tag{2}$$

$$TN=\frac{N_{L\to L}}{N_{L\to L}+N_{L\to M}} \tag{3}$$

$$FN=\frac{N_{M\to L}}{N_{M\to M}+N_{M\to L}} \tag{4}$$

$$P=\frac{N_{M\to M}}{N_{L\to M}+N_{M\to M}} \tag{5}$$

$$R=TP \tag{6}$$

$$F_\beta=\frac{(\beta^2+1)PR}{\beta^2P+R} \tag{7}$$

$$F_1=\frac{2PR}{P+R} \tag{8}$$

$$ACC=\frac{N_{L\to L}+N_{M\to M}}{N_{L\to L}+N_{L\to M}+N_{M\to L}+N_{M\to M}} \tag{9}$$

$$CER_n=\frac{\sum_{i=1}^{n}N_{M\to L}}{\sum_{i=1}^{n}(N_{M\to L}+N_{M\to M})} \tag{10}$$

此外，在实时恶意网页识别系统中，累计错误率(Cumulative Error Rate，CER)是十分重要的功能评价指标之一．累计错误率是一个随时间变化的累计值，第 n 天的累计错误率可由式(10)计算得到．

对于一些实时或者近实时系统[35-36]来说，快速识别和发现新的恶意网页十分重要．因此，Invernizzi 等人[33]以原有的恶意网页作为基础，寻找和识别新的恶意网页．此时，原有的评价指标不再适用．于是，文献[33]提出了一类新的评价指标：毒性(toxicity)和扩展度(expansion)．毒性是指所分析网页中真正恶意网页所占的比例．毒性越高，识别单个网页所需要消耗的计算资源越少．扩展度是指从一个恶意网页出发可以发现的新恶意网页数目．扩展度越高，意味着原有恶意网页的利用越充分．对同一恶意网页识别系统而言，毒性和扩展度一般存在着此消彼长的关系，即毒性越高，扩展度越低．

常用的性能指标主要包括处理一个网页所需要的时间、单位时间内网页吞吐量等．

4　恶意网页识别研究进展

目前，恶意网页识别的方法主要包括 4 类：基于黑名单技术的识别方法、基于启发式规则的识别方法、基于机器学习的识别方法以及基于交互式主机行为的识别方法．

4.1　基于黑名单技术的识别方法

黑名单是一份包含恶意网页 URL、IP 地址或者关键词信息的列表．通过使用黑名单技术，人们可以准确识别已被确认的恶意网页，从而降低误报率 FPR．根据包含信息种类的不同，黑名单可以细分为 URL 黑名单、IP 地址黑名单以及 DNS 黑名单等．黑名单技术实现简单，使用方便，因而广泛应用于 Google Safe Browsing③、Malware Domain List⑧及 PhishTank⑨等实际项目和系统中．在实际应用中，黑名单技术常常需要与人工检查、蜜网技术[37]等其他技术配合使用．

以 Google Safe Browsing API③为例，它根据 Google 提供的持续更新的 URL 列表，允许用户检查特定 URL 是否存在于这个列表上，以判断其是否为网络钓鱼或恶意软件．而基于域名的黑名单(Malware Domain List⑧)则主要依据域名或者 IP 地址信息识别和过滤对特定网站或网址的访问行为．此外，PhishTank⑨为人们提供了一个自愿提交和共享钓鱼网页网址的开放平台．人们可以依据

⑧ Malzilla. Malware Domain List. http://www.malware-domainlist.com，2014.10.20

⑨ OpenDNS. PhishTank. http://www.phishtank.com，2014.10.20

PhishTank 提供的列表主动过滤钓鱼网址，从而保障网络安全.

然而，黑名单仅能识别已经发现的恶意网页，而不能正确识别之前未出现的恶意网页，从而容易引起漏判. 为了改善漏判情况，Prakash 等人[38]针对黑名单技术提出了一种改进方法 PhishNet. PhishNet 将已经发现的钓鱼 URL 作为先验知识，通过 URL 分解和相似性计算来识别和发现新的钓鱼网页. 通过这种方式，PhishNet 扩展了黑名单的使用范围，有助于识别部分未出现的恶意网页，但它的识别能力依赖于原有黑名单集合的规模，并存在时间开销随黑名单规模扩大而线性增长的缺点.

除了上述漏判和时间开销大的问题，黑名单还存在更新时效性低的缺点. 在使用黑名单技术的过程中，当发现疑似恶意网页时，首先由综合检查技术进行分析确认，然后依据实际更新策略进行分发和部署. 综合检查技术的水平决定了新恶意网址的确认时间，而更新策略决定了每次更新内容的实际生效时间. 由于网络和计算资源的限制等原因，每次更新的时效性难以保证. 以钓鱼网站为例，根据 Sheng 等人[39]的研究，约有 63% 的网络钓鱼行为在最初的 2 h 内就结束了，而 47%～83% 的钓鱼网址在发现 12 h 后才能录入黑名单. 由此可见，黑名单更新时效性低的缺点将在很大程度上限制黑名单技术的实际使用效果.

4.2 基于启发式规则的识别方法

为了克服黑名单机制所产生的漏判等缺点，研究人员设计并实现了基于启发式规则的恶意网页识别方法. 这类方法的工作原理是依据恶意网页之间存在的相似性设计和实现启发式规则，进而发现和识别恶意网页. 不同于黑名单依靠精确匹配完成恶意网页识别，基于启发式规则的方法不需要提前了解恶意网页的网址等信息，就可以依据现有规则识别部分未出现（未识别）的恶意网页. 因此，它在一些主流浏览器上（包括火狐浏览器、IE 浏览器等）得到广泛应用，并且常以浏览器安全插件的形式出现.

2004 年，Chou 等人[40]开发出一套浏览器插件 SpoofGuard. SpoofGuard 部署在客户端，它的工作原理是：首先依据钓鱼网页常见情况建立启发式规则，从而对 HTML 网页及其 URL（包括用户输入信息、链接关系及可疑的网址信息及图片信息等）进行检测.

2007 年，Zhang 等人[41]研发出一套针对 IE 的工具条 Cantina，通过分析网页的词频-逆向文档频率（TF-IDF），搜索返回结果及其他统计信息（例如网页中是否包含特殊字符-或@及 URL 中点的个数等），建立启发式规则以判别当前网页是否是恶意网页.

基于启发式规则的方法往往假设对于某些恶意网页，其统计特征（例如链接关系、网页内容是否包含关键词等）是唯一的，可以作为规则对恶意网页和良性网页进行区分. 但是，对于大规模网页分类而言，简单的特征统计和启发式规则方法已经无法满足需求，主要体现在以下两个方面：

（1）误报率高. 由于采用启发式规则的模糊匹配技术，这类方法将大大提升良性网页的误判概率. 因此，相较于黑名单方法，启发式规则的识别方法误报率较高.

（2）规则更新难，依赖于领域知识. 由于启发式规则是通过对已有恶意网页的特征统计或人工总结得到的，因此这些规则依赖于对应的领域知识，因此更新困难.

4.3 基于机器学习的识别方法

针对基于启发式规则识别方法存在的误报率高和规则更新难的问题，研究人员进一步提出了更加系统的基于机器学习的识别方法.

这类方法首先将恶意网页识别看作是一个文本分类或聚类的问题，然后运用相应的机器学习算法（例如 k-means、DBSCAN、k-NN、C4.5 及 SVM 等）进行识别. 目前，用于恶意网页识别的机器学习方法主要包括无监督方法和有监督方法.

4.3.1 无监督机器学习方法

无监督机器学习方法又称聚类方法. 这类方法首先将 URL 数据集划分为若干簇，使得同一簇的数据对象之间相似度较高，而不同簇的数据对象之间的相似度较低. 然后，通过构造和标记数据集中的簇来区分恶意网页和良性网页. 其具体分类过程如图 3 所示，主要由特征提取、聚类、簇标记和网页判别等步骤组成.

2010 年，Liu 等人[42]以链接关系、关键词排序关系、文本相似性关系、层次相似性关系等作为统计特征，利用无监督学习算法 DBSCAN[43]（Density-Based Spatial Clustering of Applications with Noise）对钓鱼网页的攻击目标进行识别. 实验结果表明，该方法可以识别 91.44% 钓鱼网页的攻击目标，并将误报率控制在 3.4% 以内.

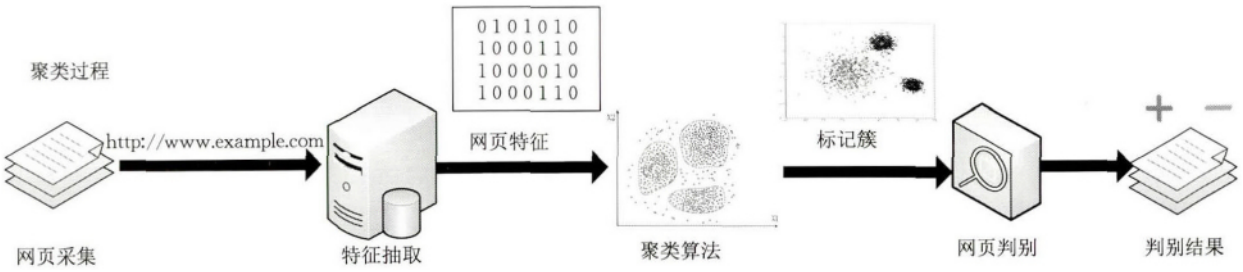

图 3　基于聚类算法的恶意网页识别过程

4.3.2　有监督机器学习方法

有监督机器学习方法又称分类方法. 研究人员通过引入网页信誉库的方式构造 URL 标注集，从而利用现有的分类算法为识别恶意网页提供了一类新思路. 具体来说，他们根据已标记样本的特点构造分类规则或分类器，进而将未知类型的样本映射到给定类别中的一个. 分类算法的工作流程如图 4 所示，它主要包括两个步骤：

(1) 训练. 根据提供的训练数据集及特征集合构造一个分类模型.

(2) 预测. 依据训练过程产生的分类模型对未知样本的类别进行预测.

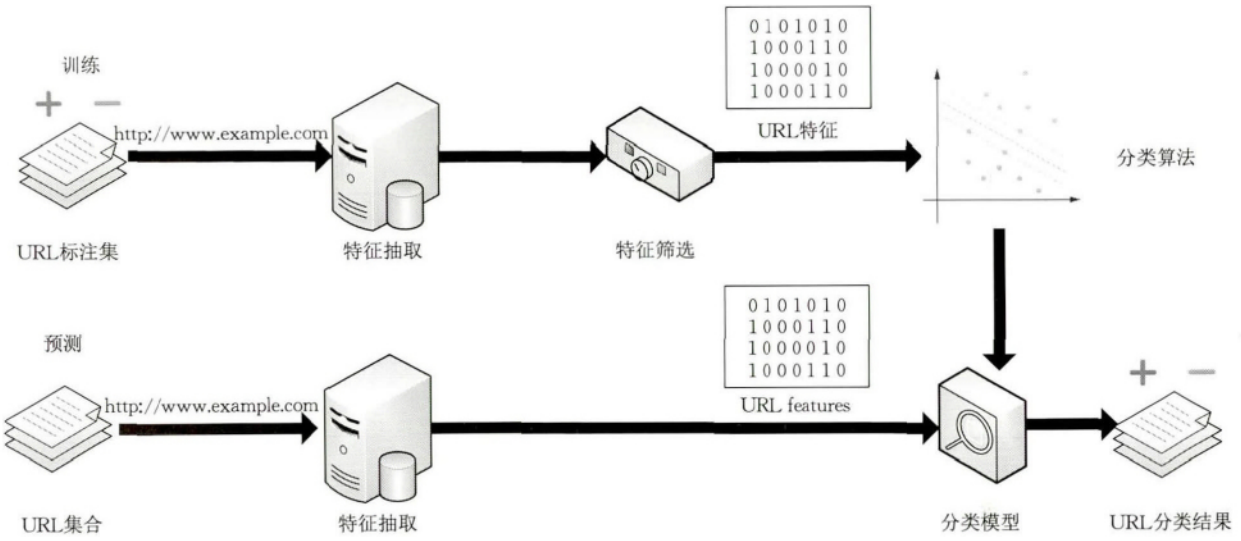

图 4　基于分类算法的恶意网页识别过程

其中用于训练和测试的数据需要具有相同的特征集. 此外，训练集应当标记好网页所属的类别，并对特征向量进行归一化，其标准格式如表 2 所示. 其中，M 代表恶意网页，L 代表良性网页，归一化后每个特征的取值范围一般为[0,1].

表 2　网页特征集合示例

序号	类别	特征集示例		
		是否在黑名单	TTL 值	URL 长度
1	M	1	0.2	0.5
2	L	0	0.5	0.1
3	L	0	0.3	0.3
4	M	0	1.0	0.8
5	L	0	0.8	0.2

3.2 节详细介绍了机器学习算法所使用的特征集. 本节将侧重于从分类模型选择方面介绍分类算法在恶意网页识别中的应用情况. 分类算法按照实时性不同分为离线分类算法和在线分类算法. 其中，常用的离线分类算法主要有决策树、贝叶斯、支持向量机(Support Vector Machine，SVM)、逻辑回归(Logistic Regression，LR)等. 而在线分类算法主要包括 PA(Passive Aggressive)、CW(Confidence Weighted)算法以及 AROW(Adaptive Regularization of Weights)等.

贝叶斯分类算法首先假设网页的每个特征之间相互独立. 在已知网页特征向量的先验概率的情况下，它依据贝叶斯公式对其后验概率进行计算. 通过比较不同网页的后验概率和预先设定好的阈值来判别网页是否属于恶意网页. Ma 等人[5]使用朴素贝叶斯分类器在多个公开数据集上进行检测，其恶意网页识别的准确率在 94%以上. 贝叶斯分类算法相对简单，可解释性强，分类速率快. 它的局限性在于假设条件较强，在恶意网页识别领域，特征之间往往

并不独立.

和贝叶斯分类算法不同,支持向量机利用非线性变换和结构风险最小化原则来提高分类器泛化能力,具有良好的分类准确率和稳定性.此外,它无需满足特征相互独立的假设,并且在统计样本量很少的情况下也能获得较高的准确率.基于 SVM 算法,Huang 等人[44]提出了一套新的钓鱼网页识别系统,并达到了 99%的识别准确率.SVM 方法[45]存在的主要缺陷在于它的可解释性较差,在高特征维度的情况下分类速率较低.此外,SVM 方法的分类结果对训练集实际分布情况以及参数配置情况较为敏感,容易产生"过拟合"现象.

逻辑回归分类器依据特征向量到超平面决策边界的距离来进行分类.LR 分类器首先根据已知的标注数据集进行训练,学习到一组权值 $w_0, w_1, \cdots, w_m$.当测试网页到达时,将这组权值与其特征向量线性加和,依据式(11)求出 z 值:

$$z = w_0 + w_1 \times x_1 + \cdots + w_m \times x_m \quad (11)$$

其中 $x_1, x_2, \cdots, x_m$ 是样本数据的各个特征,特征向量的维度为 m.

然后通过 sigmod 函数[36]对其进行归一化,以判别该网页属于恶意网页的概率.Lee 等人[36]从 URL 的重定向链接关系出发,引入重定向链接长度、入口 URL 的出现频率等信息作为特征,在 LR 分类器的基础上提出了一种名为 WarningBird 的近实时可疑 URL 监测系统,对可疑 URL 进行识别,取得了 91.90%的准确率.

逻辑回归分类器是线性分类器,因此它分类速率快,适用于大规模数据集.此外,它还具有准确率高,可解释性好的优点.它的缺陷主要是当输入数据存在偏差的情况下分类器不收敛.

表 3 对比了一些主流的离线分类器的分类效果和适用范围.离线分类算法通过对整个数据集进行分析,得到全局最优的分类策略.然而,这类算法无法对数据流形式的训练集进行学习,并且处理的数据集规模始终受到内存大小的限制.

表 3 常用离线分类器的比较

分类模型	分类速率	准确率	可解释性	数据集规模	局限性
朴素贝叶斯	高	低	好	大	假设特征间相互独立(往往不成立)
支持向量机	低	高	差	小	分类结果对训练集的分布以及参数配置较为敏感
逻辑回归	高	高	好	大	输入数据存在偏差可能导致分类器不收敛

为了满足恶意网页识别的实时性要求,研究人员引入在线分类算法,每次只对一条数据进行分析,动态调整参数以适应标注数据流的实时变化情况.常用的在线分类算法主要包括被动攻击型算法(Passive Aggressive algorithm, PA[46])、置信度加权算法(Confidence-Weighted algorithm, CW[47])和权重自适应正则化算法(Adaptive Regularization of Weights algorithm, AROW[48]).

PA 算法及其变种的思想是通过引入参数 T_t 和松弛变量 C 的概念来减少对分类模型中参数权值的调整.它的优点是能减少错误分类的数目,并且适用于不可分的噪声情况.而置信度加权算法 CW 认为每个学习参数都有信任度,可以用参数向量的高斯分布表示.相比于信任度大的参数,信任度小的参数更需要学习,所以会得到更频繁的修正机会.Ma 等人[6]收集了网页的 WHOIS 信息、DNS 信息、IP 地理位置信息以及 blacklist 信息作为特征,分别用 PA 算法和 CW 算法对可疑网页进行识别.结果显示,CW 算法的实验效果更好,它可以将累计错误率 CER 控制在 1%~2%.此后,Crammer 等人[48]对 CW 算法进行改进,提出了 AROW 分类器.这种分类器的优点是抗噪性强.Le 等人[16]在 AROW 算法的基础上,提出了用于识别钓鱼网页的系统 PhishDef.实验证明,AROW 算法在噪音 10%~30%的情况下,累计错误率依然可以保持在 10%以内.

4.4 基于交互式主机行为的识别方法

当访问恶意网页时,可能会出现安装恶意软件或者执行恶意脚本的情况.这时,可以结合虚拟化技术和蜜罐技术对恶意网页进行识别.此类方法的工作原理是:使用蜜罐技术,将虚拟主机作为诱饵,访问待检测网页,通过监测访问后的主机动态行为(例如创建新进程、改变注册表及下载文件等),判断该网页是否是恶意网页.根据使用系统的不同,蜜罐技术可以细分为基于模拟的低交互式蜜罐[49]和基于真实系统的高交互式蜜罐[50].诸葛建伟等人[13]对此有详细介绍,在此不作详述.

4.5 识别方法小结

4.1 节~4.4 节主要介绍了 4 类恶意网页识别技术,并结合已有的研究成果对其进行分析.依据上述分析结果,表 4 从识别方法、部署位置及评价指标等方面对这几类识别方法进行归纳总结、定性分析及横向比较.而表 5 则从识别特征及识别方法两种维度对典型的恶意网页攻击形式进行了分析和总结.

表 4　典型识别方法分类

典型工作	识别方法				部署位置			主要评价指标		
	黑名单	启发式规则	机器学习	主机行为	服务器端	网关	客户端	漏报率	误报率	分类速率
SmartScreen②	√	√					√	高	低	快
备注：SmartScreen 是基于 IE 浏览器的筛选器. 它根据 Microsoft 提供的持续更新的 URL 列表及当前网页分析到的特征，判断其是否为网络钓鱼或恶意软件，从而对用户给出提示.										
Google Safe Browsing③	√						√	高	低	快
备注：根据 Google 提供的持续更新的 URL 列表，允许用户检查特定 URL 是否存在于这个列表上，以判断其是否为网络钓鱼或恶意软件.										
PhishTank⑨	√				√	√		高	低	快
备注：PhishTank 提供了一个自愿提交和共享钓鱼网址的开放平台，方便人们查询和识别钓鱼网页.										
PhishNet[38]	√	√					√	中	低	快
备注：拓宽了黑名单的识别范围，减少了漏报率. 将已经发现的钓鱼 URL 作为先验知识，通过 URL 分解和相似性计算来识别和发现新的钓鱼网页. 缺点是可能引起较大的带宽消耗.										
SpoofGuard[39]		√					√	低	高	一般
备注：对 HTML 网页及其 URL(包括用户输入信息、链接关系、可疑的网址信息及图片信息等)进行检测. 可以识别一些尚未收录的恶意网页. 但依赖领域知识，规则更新困难，且容易产生误判.										
Cantina[41]		√					√	低	高	一般
备注：依据网页的词频-逆向文档频率(TF-IDF)，搜索返回结果及其他统计信息识别钓鱼网页. 由于部分特征依赖于搜索引擎的返回结果，因此会产生较大网络延迟，影响其分类速率.										
Automatic detection of phishing target[42]			√				√	低	低	慢
备注：采用链接关系、关键词排序关系、文本相似性关系、层次相似性关系等作为统计特征，依据聚类算法 DBSCAN 对钓鱼网页的攻击目标进行识别. 这些特征的抽取需要依赖网页内容和搜索结果，因此影响其分类速率.										
BeyondBlacklist[5]	√		√				√	低	低	快
备注：Ma 等人[5]在特征上主要抽取了 WHOIS 信息、DNS 信息及词汇特征. 在分类方法上，分析了贝叶斯、支持向量机及逻辑回归三类分类器在恶意网页识别问题的分类速率.										
WarningBird[36]			√		√			低	低	快
备注：Lee 等人[36]从 URL 的重定向链接关系出发，提出了基于 LR 分类器的近实时恶意 URL 识别系统. 它的优点是分类速率快，吞吐能力强.										
Identifying suspicious URLs[6]			√		√	√		低	低	快
备注：Ma 等人采用词汇特征和主机特征，对比了多种在线分类器(例如 PA 及 CW 等)对恶意 URL 的识别效果. 实验证明，在累积错误率方面，CW 算法的效果优于 Perceptron 算法和 PA 算法以及离线分类算法.										
PhishDef[16]			√			√	√	低	低	快
备注：Le 等人基于抗混淆的词汇特征和 AROW 算法，提出了一种高准确率、轻量级的钓鱼网页识别系统. 实验证明，在抗噪方面，AROW 算法明显优于 CW 算法.										
HosTaGe[49]				√			√	低	低	慢
备注：HosTaGe 是一种工作于移动端设备的低交互式的便携式蜜罐系统，主要用于检测无线网络中的恶意软件.										
High interaction honeypot[50]				√	√	√		低	低	慢
备注：Nicomette 等人采用高交互式的蜜罐系统，主要研究经 SSH 服务登陆宿主的入侵行为，重点分析攻击者成功获得系统权限后的行为，填补了该方面研究的空白.										

表 5　恶意网页攻击形式分类

攻击形式		钓鱼网页	恶意软件下载
识别特征	主机信息	√	√
	URL 信息	√	√
	网页内容	√	√
	跳转关系	√	√
	注册表变化	N/A	√
	文件变化	N/A	√
识别技术	黑名单	SmartScreen②，Google Safe Browsing③，PhishTank⑨	SmartScreen②，Google Safe Browsing③
	启发式规则	PhishNet[38]，SpoofGuard[39]，Cantina[41]	N/A
	机器学习	Automatic detection of phishing target[42]，BeyondBlacklist[5]，WarningBird[36]，Identifying suspicious URLs[6]，PhishDef[16]	BeyondBlacklist[5]，WarningBird[36]，Identifying suspicious URLs[6]
	主机行为	N/A	HosTaGe[49]，High interaction honeypot[50]

5　恶意网页识别面临的挑战

随着互联网的迅速发展、用户规模的不断扩大，传统恶意网页识别技术面临着一些新的挑战. 从来源上区分，这些挑战分别来自于客观环境的变化和恶意网页逃逸技术的升级. 5.1 节和 5.2 节分别介绍了这两类挑战，并就其中的技术细节进行分析和

讨论.

5.1 客观环境变化引起的挑战

5.1.1 网页规模大

互联网的迅速发展,使得网页规模由 GB、TB 级向 PB、ZB 级快速变化.巨大的网页规模对传统恶意网页识别技术提出了一些新的挑战.

首先,大量新网页的引入带来海量新特征.当使用传统机器学习算法对网页进行特征表示时,这些新特征的引入可能产生高维特征空间,并最终导致"维数灾难".因此,需要设计快速有效的特征选择方法对特征进行预先筛选或引入降维方法(如主成分分析法[51])对高维特征空间进行处理.

其次,大规模网页限制了资源消耗大的识别方法的应用范围.一些已有的恶意网页识别方法(例如基于主机行为的方法[13])需要消耗较多资源来分析主机行为或检测页面内容以判断是否为恶意网页.这些方法的准确率和召回率高,但检测时间和资源消耗较多.随着网页规模的扩大,这些方法的局限性日益明显.当使用这些方法时,往往需要结合快速过滤器[33]预先排除其中大多数良性网页.

最后,大规模网页对部署在网关和客户端位置的检测工具的性能提出了更高要求.随着互联网带宽的普遍增长及网页规模的不断扩大,网关和客户端流量逐步呈现出复杂化、多样化的趋势.如何在复杂网络流量中准确识别恶意网页并及时做出响应处理,需要研究人员结合其他技术(例如高性能网包处理技术等)进行深入研究[52].

5.1.2 数据集不均衡

在海量的网页中,恶意网页识别如同大海捞针,只有极少数网页最终被确认为恶意网页.Google 的统计报告③显示,Google 安全浏览器每天检查数以亿计的网页,仅发现几千个不安全的站点.由此可见,数据集的不均衡性对传统恶意网页识别方法的准确率和效率提出了严峻挑战.在特征选取层面,在不均衡数据集产生的大量特征中,不同特征的识别效果并不相同.为了提升识别效率,研究人员设计新的识别方法,依据少量特征快速过滤多数良性网页[53].在识别方法层面,不均匀的数据集启发研究人员突破传统方法的思路,寻找新的识别方法.例如,Invernizzi 等人[33]充分利用搜索引擎查询结果,从恶意网页出发寻找相似或相近的网页,提升恶意网页所占比例,并最终提升识别效率.

5.1.3 网页传播途径多元化

从传播途径上分析,传统的恶意网页主要依靠电子邮件、即时通讯工具(IM)以及搜索引擎进行传播.近年来,随着社交网站的兴起和移动互联网的繁荣,通过社交网站和扫描"二维码"传播网页的行为逐渐增多.网页的传播途径因而呈现出多元化的发展趋势.这一方面拓展了恶意网页识别技术的应用领域和应用场景[54].另一方面,客观上对恶意网页识别技术提出了新的挑战.为此,Lee 等人[35-36]从重定向关系出发提出了一个近实时检测系统,用于检测在社交网站上传播的可疑 URL.

在特征选取层面,这些新的应用场景的出现,催生了更多的网络特征,丰富了特征的选择范围.在识别方法层面,多元化的网络传播途径对识别检测系统的实时性提出了更高的要求.

5.2 恶意网页逃逸技术的升级

围绕恶意网页的攻防博弈一直在持续.本文第 4 节主要介绍了恶意网页识别技术.为了绕过这些技术的检测,攻击者常采用环境探测+动态加载、混淆免杀、人机识别、网页加密等技术手段来躲避检测与追踪.

5.2.1 环境探测+动态加载技术

随着互联网的发展,客户端浏览环境(例如浏览器版本及插件版本等)呈现出多样性,不同浏览环境包含的漏洞也不同.为了在提高恶意网页的攻击效率和成功率的同时保持攻击的隐蔽性,攻击者采用了"探测页面+攻击脚本"的"环境探测+动态加载"模式:即首先使用探测页面,对客户端浏览器版本、插件版本进行探测,然后决定是否使用动态加载技术(例如 DOM API 等)加载攻击脚本.其中,根据攻击者资源规模的不同,可以将这种攻击细分为单攻击脚本[55]和多攻击脚本[56],这里不再赘述.

5.2.2 URL 混淆技术

攻击者常对恶意网页的 URL、页面内容(攻击脚本)进行各种混淆、加密,以改变、消除其原有特征,以躲避特征扫描工具的识别.常用的混淆方式包括以下 4 类:

(1)在字符串中填充大量垃圾字符.

(2)改变编码方式.例如,采用十六进制编码、Unicode 编码及 escape 函数编码.

(3)使用 IP 地址代替域名.

(4)使用随机的 URL 参数.

此外,恶意网页常常综合使用上述混淆方式以增强隐蔽性,提高躲避检测的成功率.

5.2.3 人机识别技术

如 3.1 节介绍,在网页收集阶段,防御方常常采

用爬虫技术对网页进行主动采集. 为了躲避防御方的主动采集，攻击者常常采用一些人机识别(Web Robot Detection)技术[12]对访问者身份进行在线判定. 当认定客户端是人工浏览行为后，攻击者再执行进一步的攻击动作，否则推送事先准备好的良性网页.

其中，在线判定的方法可以分为基于图灵测试和基于离线分析结果两类. 其中基于图灵测试[57]的方法通过在线用户答题的方式(例如要求用户输入验证码)判断用户是否是爬虫；而基于离线分析结果按照原理可分为语法日志分析、流量模式分析和分析模型训练 3 类. 常见的识别依据包括：检查 user-agent 字段[58]、检测有无读取 robots.txt 文件的行为[59]等.

人机识别技术的使用，大大增强了恶意网页的隐蔽性，并对依赖主动采集方式的恶意网页识别方法提出了严峻的挑战. 对抗此类逃逸技术的防御技术一般围绕拓宽采集方式展开. 但由于采用其他采集方式受环境和资源的限制较多，因此有待深入研究.

5.2.4　网页加密技术

一些攻击者们开始模仿正常的在线服务网站对其网页采用 SSL 协议和 HTTPS 加密服务. 一方面，采用加密服务的网页更容易取得用户信任，提高攻陷可能性；另一方面，加密恶意网页隐藏网址信息和页面内容，可以帮助逃避部署在网关的传统识别系统的检测. 趋势科技的统计数据显示，2010 年至 2014 年间使用 HTTPS 服务的钓鱼网站⑩逐年增长，从不到 1000 个站点增加到超过 4000 个站点. 这些加密服务的使用，限制了传统依赖网址信息和页面内容的检测技术的应用范围. 研究人员往往需要结合证书信息分析和检测这类恶意网页. 对抗这一逃逸技术的防御技术还比较少，有待深入研究.

5.2.5　生命周期持续缩短

部分恶意网页的生命周期持续缩短. 以钓鱼网站为例，奇虎 360 公司的统计数据⑪显示，其生存周期已经从 2011 年的平均 50 h 左右，下降到 2012 年下半年的不足 6 h. 恶意网页生命周期的缩短，对恶意网页识别的时效性提出新的挑战，并推动了在线识别技术的发展.

6　研究展望

上述挑战在为恶意网页识别工作带来新难题的同时，客观上也为恶意网页识别技术的新发展创造了新的条件和机遇. 展望未来，仍有如下研究问题值得关注和进一步探讨.

(1)针对不同应用场景的恶意网页识别方法. 随着互联网的进一步发展，特别是社交网络和移动互联网[60]的兴起和繁荣，恶意网页识别的应用场景也随之不断变化. 应用场景的变化，一方面改变了恶意网页的传播途径，另一方面对恶意网页识别的准确率和实时性[36]提出了更高要求. 为了满足这些要求，需要研究人员不断发掘识别恶意网页的特征种类，并引入新的更加适合的识别方法进行分析. 因此，这是未来恶意网页识别的可能发展方向之一.

(2)特征的比较和评测. 目前用于识别恶意网页的特征种类繁多，规模庞大. 一种或一类特征可能在某一类恶意网页的识别问题上效果显著，但不适合识别其他恶意网页. 为了更好地对不同特征进行比较与评测，一方面，需要构造比较合理的标注数据集；另一方面，需要在合理的特征比较和评测方法上进行研究. 因此，这是未来恶意网页识别的可能发展方向之一.

(3)针对隐身逃逸技术的识别和检测. 5.2 节介绍了几种攻击者常用的隐身逃逸技术. 通过使用这些技术，攻击者可以不断调整其恶意网页的外显特征，从而规避传统方法的检测. 针对特定隐身逃逸技术[61]的识别和分类既是未来恶意网页识别的重点，也是难点. 因此，这是未来恶意网页识别的可能发展方向之一.

(4)不同识别方法的融合. 不同的识别方法从不同的侧面对恶意网页进行分析和检测，选取的特征、适用的场景各有侧重. 因此，不同识别方法之间具有很强的互补性，融合各类识别方法可能是从大规模网页中识别少量恶意网页的解决途径之一.

7　结束语

恶意网页识别是信息安全领域的热点问题. 随着网络攻击技术和防御技术的不断发展，该问题一直受到研究人员的广泛关注. 针对这一问题，本文首先梳理了恶意网页的基本概念，然后介绍了恶意网页识别的研究框架、应用场景和评价指标，进而对不同类别的识别方法进行深入分析和比较，分别指出它们的优势、不足以及适用场景.

⑩ HTTPS Phishing URLs. http://www.linuxidc.com/Linux/2014-10/107558.htm, 2014.10.20

⑪ Browser Security and Development Report of China. http://zt.360.cn/report/#5, 2014.10.20

本文还重点讨论了恶意网页识别面临的新挑战，并介绍了未来恶意网页识别领域可能的研究方向.

致 谢 本文得到了国家自然科学基金委员会、国家科学技术部等机构的支持.同时，很多同行对本文的工作给予了支持和建议，在此一并表示感谢！

参 考 文 献

[1] Mahmoud K, Youssef I, Andrew J. Phishing detection: A literature survey. IEEE Communications Surveys & Tutorials, 2013, 15(4): 2091-2121

[2] Paul K, Georgia K, Hector G M. Fighting spam on social Web sites a survey of approaches and future challenges. IEEE Internet Computing, 2007, 11(6): 36-45

[3] Priya M, Sandhya L, Ciza T. A static approach to detect drive-by-download attacks on Webpages//Proceedings of the International Conference on Control Communication and Computing. Xi'an, China, 2013: 298-303

[4] Mavrommatis N P P, Monrose M A R F. All your iframes point to us//Proceedings of the 17th USENIX Security Symposium. San Jose, USA, 2008: 1-22

[5] Ma J, Saul L K, Savage S, Voelker G M. Beyond blacklists: Learning to detect malicious Web sites from suspicious URLs//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA, 2009: 1245-1253

[6] Ma J, Saul L K, Savage S, Voelker G M. Identifying suspicious URLs: An application of large-scale online learning // Proceedings of the 26th Annual International Conference on Machine Learning. Montreal, Canada, 2009: 681-688

[7] Ma J, Saul L K, Savage S, Voelker G M. Learning to detect malicious URLs. ACM Transactions on Intelligent Systems and Technology, 2011, 2(3): 1-24

[8] Canali D, et al. Prophiler: A fast filter for the large-scale detection of malicious Web pages//Proceedings of the 20th International Conference on World Wide Web. Hyderabad, India, 2011: 197-206

[9] Thomas K, et al. Design and evaluation of a real-time URL spam filtering service//Proceedings of the IEEE Symposium on Security and Privacy. Oakland, USA, 2011: 447-462

[10] Yadav S, Reddy A K K, Reddy A L, et al. Detecting algorithmically generated malicious domain names//Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement. New York, USA, 2010: 48-61

[11] Kolbitsch C, Livshits B, Zorn B, et al. Rozzle: De-cloaking internet malware//Proceedings of the IEEE Symposium on Security and Privacy. San Francisco, USA, 2012: 443-457

[12] Zhang Hui-Lin, Zou Wei, Han Xin-Hui. Drive-by-download mechanisms and defenses. Journal of Software, 2013, 24(4): 843-858(in Chinese)
(张慧琳，邹维，韩心慧. 网页木马机理与防御技术. 软件学报，2013，24(4)：843-858)

[13] Zhuge Jian-Wei, Tang Yong, Han Xin-Hui, Duan Hai-Xin. Honeypot technology research and application. Journal of Software, 2013, 24(4): 825-842(in Chinese)
(诸葛建伟，唐勇，韩心慧，段海新. 蜜罐技术研究与应用进展. 软件学报，2013，24(4)：825-842)

[14] Xiong H, Malhotra P, Stefan D, et al. User-assisted host-based detection of outbound malware traffic//Proceedings of the International Conference on Information and Communications Security. Beijing, China, 2009: 293-307

[15] Eshete B, Villafiorita A, Weldemariam K. Binspect: Holistic analysis and detection of malicious Web pages. Lecture Notes of the Institute for Computer Sciences Social Informatics & Telecommunications Engineering, 2013, 106: 149-166

[16] Le A, Markopoulou A, Faloutsos M. PhishDef: URL names say it all//Proceedings of the 30th IEEE International Conference on Computer Communications. Shanghai, China, 2011: 191-195

[17] Shahriar H, Zulkernine M. Mutec: Mutation-based testing of cross site scripting//Proceedings of the ICSE Workshop on Software Engineering for Secure Systems. Vancouver, Canada, 2009: 47-53

[18] Whittaker C, Ryner B, Nazif M. Large-scale automatic classification of phishing pages//Proceedings of the 17th Annual Network & Distributed System Security Symposium. San Diego, USA, 2010: 1-14

[19] Rahman M S, Huang T K, Madhyastha H V, et al. Efficient and scalable socware detection in online social networks// Proceedings of the 21th USENIX Security Symposium. Bellevue, USA, 2012: 663-678

[20] Chou Li-Der, Zheng He, et al. Design and implementation of content-based filter system on embedded linux home gateway //Proceedings of the 14th International Conference on the Advanced Communication Technology. Seoul, Korea, 2012: 1046-1051

[21] Fette I, Sadeh N, Tomasic A. Learning to detect phishing emails//Proceedings of the 16th International Conference on World Wide Web. Banff, Canada, 2007: 649-656

[22] Seifert C, Welch I, Komisarczuk P, et al. Identification of malicious Web pages through analysis of underlying DNS and Web server relationships//Proceedings of the 33rd IEEE Local Computer Networks Conference. Montreal, Canada, 2008: 935-941

[23] Spirin N, Han Jia-Wei. Survey on Web spam detection: Principles and algorithms. ACM SIGKDD Explorations Newsletter, 2012, 13(2): 50-64

[24] McGrath D K, Gupta M. Behind phishing: An examination of phisher modi operandi//Proceedings of the USENIX Workshop on Large-Scale Exploits and Emergent Threats. San Jose, USA, 2008: 1-8

[25] Liang Bin, Huang Jian-Jun, Liu Fang, et al. Malicious Web pages detection based on abnormal visibility recognition// Proceedings of the International Conference on E-Business and Information System Security. Wuhan, China, 2009: 1-5

[26] Hallaraker O, Vigna G. Detecting malicious JavaScript code in Mozilla//Proceedings of the 10th IEEE International Conference on Engineering of Complex Computer Systems. Shanghai, China, 2005: 85-94

[27] Kals S, Kirda E, Kruegel C, et al. SecuBat: A Web vulnerability scanner//Proceedings of the 15th International Conference on World Wide Web. New York, USA, 2006: 247-256

[28] Zhou Li, Alrwais S, Xie Ying-Lian, et al. Finding the linchpins of the dark Web: A study on topologically dedicated hosts on malicious Web infrastructures//Proceedings of the IEEE Symposium on Security and Privacy. San Francisco, USA, 2013: 112-126

[29] Li Zhi-Yong, Ran Tao, Cai Zhen-He, Zhang Hao. A Web page malicious code detect approach based on script execution //Proceedings of the 5th International Conference on Natural Computation. Tianjin, China, 2009: 308-312

[30] Zhang Jun-Jie, Seifert C, Stokes J W, Lee W. ARROW: Generating signatures to detect drive-by downloads// Proceedings of the 20th International Conference on World Wide Web. Hyderabad, India, 2011: 187-196

[31] Seifert C, Welch I, Komisarczuk P. Identification of malicious Web pages with static heuristics//Proceedings of IEEE Conference on Telecommunication Networks and Applications. Adelaide, Australia, 2008: 91-96

[32] Wang Tao, Yu Shun-Zheng, Xie Bai-Lin. A novel framework for learning to detect malicious Web pages//Proceedings of the International Forum on Information Technology and Applications. Kunming, China, 2010: 353-357

[33] Invernizzi L, Comparetti P M, Benvenuti S, et al. EvilSeed: A guided approach to finding malicious Web pages//Proceedings of the IEEE Symposium on Security and Privacy. San Francisco, USA, 2012: 428-442

[34] Batista G E, Prati R C, Monard M C. A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD Explorations Newsletter, 2004, 6(1): 20-29

[35] Lee S, Kim J. WarningBird: Detecting suspicious URLs in twitter stream//Proceedings of the 19th Annual Network & Distributed System Security Symposium. San Diego, USA, 2012: 1-13

[36] Lee S, Kim J. WarningBird: A near real-time detection system for suspicious URLs in twitter stream. IEEE Transactions on Dependable and Secure Computing, 2013, 10(3): 183-195

[37] Honeynet Project. Know Your Enemy: Learning about Security Threats. 2nd Edition. Boston: Addison-Wesley Professional, 2004

[38] Prakash P, Kumar M, Kompella R R, et al. PhishNet: Predictive blacklisting to detect phishing attacks//Proceedings of the 29th IEEE International Conference on Computer Communications. San Diego, USA, 2010: 1-5

[39] Sheng S, Wardman B, Warner G, et al. An empirical analysis of phishing blacklists//Proceedings of the 6th Conference in Email and Anti-Spam. Mountainview, USA, 2009: 1-10

[40] Chou N, Ledesma R, Teraguchi Y, Mitchell J C. Client-side defense against Web-based identity theft//Proceedings of the 11th Annual Network & Distributed System Security Symposium. San Diego, USA, 2004: 1-16

[41] Zhang Yue, Hong J I, Cranor L F. Cantina: A content-based approach to detecting phishing Web sites//Proceedings of the 16th International Conference on World Wide Web. Banff, Canada, 2007: 639-648

[42] Liu Gang, Qiu Bite, Liu Wen-Yin. Automatic detection of phishing target from phishing Webpage//Proceedings of the 20th International Conference on Pattern Recognition. Istanbul, Turkey, 2010: 4153-4156

[43] Ester M, Kriegel H-P , Sander J, Xu Xiao-Wei. A density-based algorithm for discovering clusters in large spatial databases with noise//Proceedings of the 2nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Portland, USA, 1996: 226-231

[44] Huang H, Qian L, Wang Y. A SVM-based technique to detect phishing URLs. Information Technology Journal, 2012, 11(7): 921-925

[45] Kolari P, Finin T, Joshi A. SVMs for the Blogosphere: Blog identification and splog detection//Proceedings of the AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs. Menlo Park, USA, 2006: 92-99

[46] Crammer K, Dekel O, Keshet J, et al. Online passive-aggressive algorithms. Journal of Machine Learning Research, 2006, 7: 551-585

[47] Blum A, Wardman B, Solorio T, Warner G. Lexical feature based phishing URL detection using online learning// Proceedings of the 3rd ACM Workshop on Artificial Intelligence and Security. Chicago, USA, 2010: 54-60

[48] Crammer K, Kulesza A, Dredze M. Adaptive regularization of weight vectors. Machine Learning, 2013, 91(2): 155-187

[49] Vasilomanolakis E, Karuppayah S, Fischer M, et al. This network is infected: HosTaGe-a low-interaction honeypot for mobile devices//Proceedings of the 3rd ACM Workshop on Security and Privacy in Smartphones & Mobile Devices. San Francisco, USA, 2013: 43-48

[50] Nicomette V, Kaâniche M, Alata E, et al. Set-up and deployment of a high-interaction honeypot: Experiment and lessons learned. Journal in Computer Virology, 2011, 7(2): 143-157

[51] Bishop C M. Pattern Recognition and Machine Learning. New York: Springer, 2006

[52] Qi Ya-Xuan, Li Jun. Theoretical analysis and algorithm design of high-performance packet classification algorithms. Chinese Journal of Computers, 2013, 36(2): 408-421 (in Chinese)
(亓亚烜, 李军. 高性能网包分类理论与算法综述. 计算机学报, 2013, 36(2): 408-421)

[53] Lin Min-Sheng, Chiu Chien-Yi, Lee Yuh-Jye, Pao Hsing-Kuo. Malicious URL filtering—A big data application// Proceedings of the IEEE International Conference on Big Data. Santa Clara, USA, 2013: 589-596

[54] Wählisch M, Trapp S, Keil C, et al. First insights from a mobile honeypot//Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication. Helsinki, Finland, 2012: 305-306

[55] Kapravelos A, Shoshitaishvili Y, Cova M, et al. Revolver: An automated approach to the detection of evasive Web-based malware//Proceedings of the 22nd USENIX Security Symposium. Washington, USA, 2013: 637-652

[56] Chen Kevin Zhijie, Gu Guo-Fei, Zhuge Jian-Wei, et al. WebPatrol: Automated collection and replay of Web-based malware scenarios//Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security. Singapore, 2011: 186-195

[57] Doran D, Gokhale S S. Web robot detection techniques: Overview and limitations. Data Mining and Knowledge Discovery, 2011, 22(1): 183-210

[58] Stevanovic D, Vlajic N, An A. Detection of malicious and non-malicious Website visitors using unsupervised neural network learning. Applied Soft Computing, 2013, 13(1): 698-708

[59] Lee J Y, Lee Y T. A framework for a research inventory of sustainability assessment in manufacturing. Journal of Cleaner Production, 2014, 79: 207-218

[60] Yang Huan, Zhang Yu-Qing, Hu Yu-Pu, Liu Qi-Xu. A malware behavior detection system of Android applications based on multi-class features. Chinese Journal of Computers, 2014, 37(1): 15-27(in Chinese)
(杨欢, 张玉清, 胡予濮, 刘奇旭. 基于多类特征的 Android 应用恶意行为检测系统. 计算机学报, 2014, 37(1): 15-27)

[61] Sha Hong-Zhou, Zhou Zhou, Liu Qing-Yun, Qin Peng. Light-weight self-learning for URL classification. Journal on Communications, 2014, 35(9): 32-39(in Chinese)
(沙泓洲, 周舟, 刘庆云, 秦鹏. 轻量级的自学习网页分类方法. 通信学报, 2014, 35(9): 32-39)

SHA Hong-Zhou, born in 1988, Ph. D. candidate. His research interests include information security, network security.

LIU Qing-Yun, born in 1980, Ph. D., senior engineer. His research interests include information security, network security.

LIU Ting-Wen, born in 1986, Ph. D., assistant researcher. His research interests include big data security analysis, algorithm design and analysis.

ZHOU Zhou, born in 1983, Ph. D., senior engineer. His research interests include network security, high-performance network.

GUO Li, born in 1969, senior engineer. Her research interests include information security, network security and data stream processing.

FANG Bin-Xing, born in 1960, Ph. D., professor, Ph. D. supervisor, member of Chinese Academy of Engineering. His current research interests include network security and information content security.

Background

With the rapid development of network technology, the malicious webpage detection has become much more important than ever in the field of network security. During recent years, a great number of researchers have paid much attention to this area, and focused on theoretical model and practical applications. However, a lot of issues still have not been addressed well.

Thus, this paper first studies the basic concept of malicious web pages and then introduces the research framework, application scenarios and evaluation index of the detection process. Furthermore, it makes in-depth analysis among different categories of detection approaches and points out their advantages, disadvantages separately. Besides, this paper also discusses the new challenges in this area, and introduces the possible research direction in this field in the future.

This work is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences (No. XDA06030200), the National Science and Technology Support Program (No. 2012BAH46B02) and the National Natural Science Foundation of China (No. 61402474).