Journal of Computer Applications
计算机应用,2014,34(6):1661 -1665
ISSN 1001-9081
CODEN JYIIDU
2014-06-10
http://www.joca.cn

文章编号:1001-9081(2014)06-1661-05
doi:10.11772/j.issn.1001-9081.2014.06.1661

OSN 中基于分类器和改进*n*-gram模型的跨站脚本检测方法

李沁蕾[1,2],王　蕊[1],贾晓启[1*]
(1.信息安全国家重点实验室(中国科学院信息工程研究所),北京 100093;　2.中国科学院大学,北京 100049)
(*通信作者电子邮箱 jiaxiaoqi@iie.ac.cn)

摘　要:针对在线社交网络中跨站脚本(XSS)攻击的安全问题,提出了一种在线社交网络恶意网页的检测方法。该方法依据在线社交网络中跨站脚本恶意代码的传播特性,提取一组基于相似性和差异性的特征,构造分类器和改进*n*-gram模型,再利用两种模型的组合,检测在线社交网络网页是否恶意。实验结果表明,与传统的分类器检测方法相比,结合了改进*n*-gram模型的检测方法保证了检测结果的可靠性,误报率约为5%。

关键词:在线社交网络;跨站脚本攻击;分类器;*n*-gram模型;检测
中图分类号:TP393.08　　**文献标志码**:A

Cross-site scripting detection in online social network based on classifiers and improved *n*-gram model

LI Qinlei[1,2], WANG Rui[1], JIA Xiaoqi[1*]
(1. *State Key Laboratory of Information Security (Institute of Information Engineering, Chinese Academy of Sciences), Beijing* 100093, *China*;
2. *University of Chinese Academy of Sciences, Beijing* 100049, *China*)

Abstract: Due to the threats of Cross-Site Scripting (XSS) attack in Online Social Network (OSN), a approach combined classifiers and improved *n*-gram model was proposed to detect the malicious OSN webpages infected with XSS code. Firstly, similarity-based features and difference-based features were extracted to build classifiers and the improved *n*-gram model. After that, the classifiers and model were combined to detect malicious webpages in OSN. The experimental results show that compared with the traditional classifier detection methods, the proposed approach is more effective and the false positive rate is about 5%.

Key words: Online Social Network (OSN); Cross-Site Scripting (XSS) attack; classifiers; *n*-gram model; detection

0　引言

近几年来,已拥有大量用户的在线社交网络(Online Social Network, OSN),其用户数量仍在上升[1]。OSN 在全球拥有的巨大用户数量和其隐藏的潜在经济利益,引起了许多黑客们的关注和兴趣,导致针对 OSN 平台的网络攻击层出不穷[2],几种常见的网络攻击包括:恶意软件、病毒、骗局、欺诈、盗窃等。跨站脚本(Cross-Site Scripting, XSS)攻击是威胁 OSN 平台安全性的一种常见且极具危害性的攻击形式[3]。OSN 应用中,用户拥有自己的关系网络,网络内部的节点交互频繁,一旦关系网络中有一个节点受到 XSS 攻击,这个关系网络中任意其他节点用户受到攻击的概率将远高于普通网络中的节点。因此,为了控制 XSS 攻击在 OSN 中的传播,有效的检测手段是亟待解决的问题。

本文针对在线社交网络中感染了 XSS 恶意代码的网页,提出了一种检测方法,主要工作如下:

1)依据在线社交网络与传统网络应用的相似性和差异性,从网页中提取一组可用于网页分类识别的特征,并构造分类器;

2)根据*n*-gram模型在安全领域的应用,提出一种改进*n*-gram模型,此模型可以利用特征向量,识别恶意网页;

3)结合分类器和改进*n*-gram模型,实现两种检测方案,达到对 OSN 中感染了 XSS 恶意代码的网页进行检测的目的;

4)实验中,构造了模拟 OSN 中 XSS 蠕虫传播的样本,实验结果证明了方法的有效性。

1　相关工作

微软研究院 Livshits 等[4]实现了 Spectator 系统,该系统通过在用户上传的数据中注入标签和用户 IP(Internet Protocol)来构造数据在社交网络中的传播图,以传播图的直径为检测依据判断社交网络中是否存在 JavaScript 蠕虫。美国西北大学网络与安全技术实验室 Cao 等[5]提出的 PathCutter,是一种用于检测并阻断社交网络中蠕虫传播的方法,它主要是在网络中 XSS 蠕虫传播路径的两个重要步骤上进行阻拦:一是在 DOM(Document Object Model)访问客户端的不同视图时;二是在向服务器提交未授权的 HTTP(Hyper Text Transfer Protocol)请求时。阻止恶意代码的传播,就能在一定程度上抑制网络危害的蔓延。Ter Louw 等[6]设计的一种结合了内容过滤和浏览器协作的工具 BLUEPRINT,从 HTML

收稿日期:2013-11-29;**修回日期**:2014-01-15。　**基金项目**:国家自然科学基金资助项目(61100228);中国科学院战略性先导专项(XDA06030601, XDA06010701);国家 863 计划项目(2012AA013101)。
作者简介:李沁蕾(1989-),女,安徽铜陵人,硕士研究生,主要研究方向:恶意代码分析检测;　王蕊(1981-),女,北京人,副研究员,博士,CCF 会员,主要研究方向:网络与信息系统安全;　贾晓启(1982-),男,北京人,副研究员,博士,CCF 会员,主要研究方向:恶意代码分析检测、虚拟化、网络和操作系统安全。

(Hypertext Markup Language)、CSS(Cascading Style Sheet)、URI(Uniform Resource Identifier)和 JavaScript 的角度对输入内容进行分析,抵御恶意的用户输入,阻止恶意脚本插入网络应用。

另一方面,分类器是一种检测 XSS 恶意代码的静态方法,它在检测过程中花费的时间成本较低,被应用于多种检测系统。由于混淆技术是制造恶意代码的常用手段,分类器方法多从代码的混淆性角度进行恶意代码检测。Likarish 等[7]从网页中提取了 65 个特征,根据 4 种分类器算法构造分类器模型,对混淆的 JavaScript 进行检测。良好的检测率证明了此方法在混淆脚本检测中的有效性。针对如何识别注入了 XSS 恶意代码的网页的问题,Nunan 等[8]提出一种基于文件内容和 URL(Uniform Resource Locator)特征进行自动分类的方法,实现对网页的分类(XSS infected 和 non-infected 两类)。

n-gram模型是一种用于预测序列中下一项元素的方法,同时,在异常检测中*n*-gram模型也有所应用。Li 等[9]描述了一种基于*n*-gram分析的恶意软件检测方法,被称作 Fileprint。Fileprint 应用 1-gram 对多种类型的文件进行分析以检测出恶意的代码,该方法在实验中证明了其有效性,并获得了良好的效率。AccessMiner 也是一种利用*n*-gram模型进行检测的方法,由 Lanzi 等[10]提出,该方法从软件的系统调用序列中建立*n*-gram模型,以检测出可能的恶意软件。检测前,构建一个训练数据的恶意软件*n*-gram模型数据库;检测时,当一个软件的*n*-gram模型中有超过 k(阈值,按照经验值确定)个*n*-gram项与数据库匹配时,该软件被检测为恶意的。

2 改进的*n*-gram模型

n-gram模型是一种用于预测一个序列中下一项的概率的语言模型,它类似于 $n-1$ 阶的马尔可夫链,这个模型可用于概率论、计算语言、通信理论等很多应用。在安全领域,*n*-gram也有一定的应用,例如,一种用于检测计算机恶意软件的*n*-gram模型。

对于一个运行在系统上的软件,为了建立起它的*n*-gram集合,首先提取软件运行时的系统调用号序列。假设软件 A 的系统调用号序列为 $a_1, a_2, a_3, a_4, a_5, a_6$,那么软件 A 的 3-gram 集定义为$\{\langle a_1, a_2, a_3\rangle, \langle a_2, a_3, a_4\rangle, \langle a_3, a_4, a_5\rangle, \langle a_4, a_5, a_6\rangle\}$。对于一组包含了若干恶意软件和善意软件的软件集,用它们的 3-gram 集合建立用于检测恶意软件的 3-gram 模型比对库。

设建模样本中包括 m 个恶意软件,它们所有的 3-gram 集合为 M,b 个善意软件,它们的 3-gram 集合为 B,取 M 和 B 的差集$(M-B)$为检测的比对库。对一个待检测的软件,它的 3-gram 模型含有 p 个 3-gram,将 p 个 3-*gram* 与比对库中项进行匹配,当匹配的项数超过 k(k 为依据经验设定的阈值)时,该软件被检测为恶意的,否则为善意软件。

n-gram模型是基于序列构造的模型,为了避免顺序相关性对模型的影响,本文建立一种改进*n*-gram模型。改进*n*-gram模型的建立基于一组相关但无序的数值,如:分类器输入的特征向量,向量中的项无序,但是它们之间隐藏了关联性和依赖性(如:网页中长脚本字符串个数与最大脚本字符串长度相关)。

对一个特征向量$\langle a_1, b_1, a_2, b_2, a_3, b_3\rangle$,假设特征向量中 a_1 与 b_1,a_2 与 b_2,a_3 与 b_3 相关联,即 a_1 的值会影响 b_1 的取值范围,并且 a_1、a_2、a_3(b_1、b_2、b_3)代表的意义不同,具有不同的取值范围。为了区分意义不同的*n*-gram组,需要在模型的每个*n*-gram中添加一项序号,因此,基于特征向量构造的改进*n*-gram模型为$\{\langle id_1, a_1, b_1\rangle, \langle id_2, a_2, b_2\rangle, \langle id_3, a_3, b_3\rangle\}$。

当改进*n*-gram模型应用于恶意网页检测时:1)建立模型的比对库。为了避免*n*-gram方法中阈值 k 的难确定性和低扩展性,本文方法同时构造了恶意样本的比对库(记作 MDB)和善意样本的比对库(记作 BDB),利用待检测样本的改进*n*-gram模型与 MDB 匹配的个数和与 BDB 的匹配个数的差异判断是否恶意。2)检测样本。对一个待检测样本,首先建立它的改进*n*-gram模型,接着从 MDB 和 BDB 中分别计算匹配的项数 m 和 b,当 $m>b$ 时,待检测样本被识别为恶意;否则,为善意。

3 特征提取

特征是判断样本类别的重要依据,特征向量是样本的一组特征集合。在分类器和改进*n*-gram模型中,特征向量是检测恶意网页的重要依据。特征提取,即是从网页源码文件中,提取一组量化的特征,作为恶意网页检测的依据。

3.1 基于相似性的特征

作为一种网络应用,OSN 与传统网络应用具有很多的相似性。在分析中,本文总结了一些 OSN 与传统网络的相似点:1)无论是 OSN 还是传统网络应用,其网页源码中都包含 HTML 标签、JavaScript 脚本、URL 等;2)当网页中被植入 XSS 恶意代码时,网页源码在形式上或多或少都会发生一些变化;3)应用中会使用一些新兴的网页技术,如:AJAX(Asynchronous JavaScript and XML)。根据它们的相似点,本文提取一组基于相似性的特征,如表 1 所示。

表 1 基于相似性的特征

序号	特征名称
1	DOM 修改函数出现的次数
2	字符串解码函数出现次数
3	运行字符串函数出现次数
4	AJAX 关键字次数
5	其他重要的关键字
6	出现拼接字符串的次数
7	脚本单字符串的最长长度
8	长字符串个数
9	字符串编码所占最大比例
10	URL 的最长长度
11	长 URL 个数
12	URL 混淆字符比例最大值
13	HTML 标签的最长长度
14	长 HTML 标签个数
15	HTML 标签中脚本字符串个数

表 1 描述了一组基于在线社交网络相似性的特征:1)脚本程序中存在着一些关键字(关键函数),它们在良性脚本和恶意脚本中出现的频率相差很大[11],如 DOM 修改函数、字符串编码解码函数等;2)在恶意脚本中,混淆脚本程序是一种常见的用于躲避检测器过滤的方法,被混淆的脚本从表现形式上比普通脚本更复杂、长度更长;3)被恶意代码“装饰”过的 URL,在形式上和普通 URL 有很大差异,长度较长、内容可读性较差等;4)HTML 标签是网页中藏匿恶意脚本的最佳位

置,因此较长的带有脚本的 HTML 标签的可疑性较高。

3.2　基于差异性的特征

OSN 与传统网络应用相比具有独特性。XSS 恶意代码在 OSN 中的传播有别于传统网络中的传播,最直观的特点是:OSN 拓扑图中节点的高聚集度和较小的节点间平均最短距离,导致 OSN 中的 XSS 恶意代码传播速度远高于其在传统网络中的传播速度[12]。与 2003 年的计算机病毒 Blaster 在 20 h 内感染 336 000 相比,社交网络 XSS 蠕虫 Samy 在 20 h 内感染了 1 000 000 个用户[13-14]。

代码的传播速度是一个判断代码是否恶意的有效特征。为了量化并能快速地提取这一特征,本文规定单位时间内可疑内容在服务器发送的网页中出现的次数为可疑内容的传播速度。对于恶意代码,它的快速传播导致服务器发往客户端的网页中出现可疑代码的频率增大。如果社交网络中存在 XSS 恶意代码,那么单位时间内,这部分代码出现在服务器生成的网页中的次数将远高于其他内容。

基于 OSN 中恶意代码传播速度的差异性,本文提出了三个相关的特征:可疑 JavaScript 脚本字符串出现次数、可疑 HTML 标签出现次数、可疑 URL 出现次数,分别估计可疑内容在 OSN 应用中的传播速度。

4　OSN 中 XSS 的检测

本文提出了利用分类器和改进n-gram模型结合的方法来实现对 OSN 网页的检测。在检测方案中,将选择 ADTree (Alternating Decision Tree)[15] 作为构建分类器的分类算法。ADTree 是一种结合了 boosting 的决策树算法,它的决策规则较少且较容易解释,因此也更适用于实际的应用环境。ADTree 与常用分类算法 Naïve Bayes 和支持向量机(Support Vector Machine, SVM)相比具有以下优势:1) ADTree 可用于相互独立性低、依赖性高的特征组;2) ADTree 可处理缺省的特征向量;3) ADTree算法使得构建分类器所花费的时间较少且更加方便。

4.1　检测方案 1

在检测方案 1 中,分类器和改进n-gram模型是主辅关系。对于分类问题,尤其是二分问题,分类器是一种简洁而有效的解决办法。为了对分类器的误报率和漏报率进行改善,本文方案利用改进n-gram模型对分类器的结果进行辅助修正,从而得到一个漏报率更低的检测方案。方案的实现过程如图 1 所示。

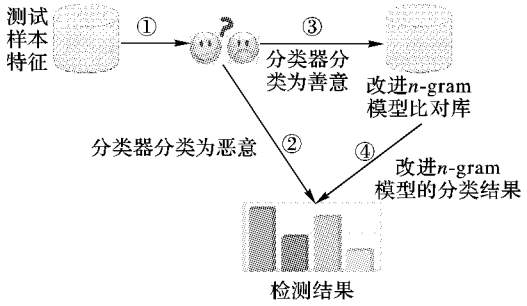

图 1　检测方案 1

检测过程如下:

1)分类器检测。首先,分类器对样本进行分类,分类器输入为样本的特征向量,输出为分类结果,记为 $result_1$($result_1$ = benign 或者 malicious)。

2)改进n-gram模型检测。改进n-gram模型对恶意网页有强标示,即改进n-gram模型如同网页的“指纹”,当网页的“指纹”在“恶意指纹库”中有记录时,该网页将被认定为恶意,因此,改进n-gram模型对恶意网页检测效果更佳。为了减少分类器产生的漏报数量,利用改进n-gram模型对分类器分类为 benign 的样本进行辅助判断,若样本的“指纹”存在于“恶意指纹库”,则网页的分类结果修正为 malicious。

4.2　检测方案 2

在检测方案 2 中,分类器和改进n-gram模型是并列关系,网页的类型由两者组合的部件决定。此方案中,设计一个具有三个子部件的决策盒,盒中的每个部件对网页的检测过程相对独立,决策盒的最终决策结果由三个子部件的检测结果共同决定。图 2 描述了方案的实现过程,具体检测过程如下:

1)决策盒的组成。决策盒内有三个子部件:第一个部件包含了分类器 A,第二个部件包含了改进n-gram模型比对库,第三个部件包含了改进n-gram模型比对库和分类器 B。其中,分类器 A 的输入为第 3 章提取的特征向量,分类器 B 比分类器 A 多了一项特征,该项特征表示样本在改进n-gram模型比对库中模型匹配“恶意指纹”的程度。

2)检测过程。决策盒的输入为待检测样本的特征向量,它利用样本的特征向量对样本进行检测,当盒中三个子部件中有两个及以上的部件判断样本为 malicious 时,样本被检测为 malicious;否则为 benign。

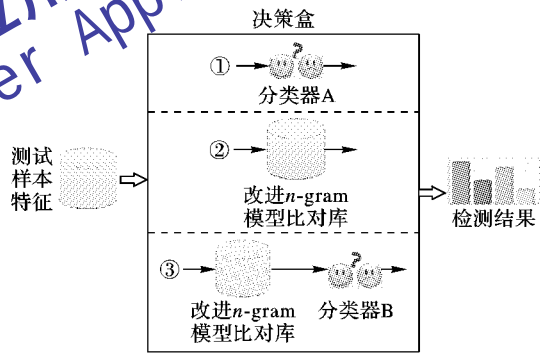

图 2　检测方案 2

5　模拟实验及分析

5.1　模拟感染 XSS 蠕虫样本

由于在真实的环境下,重现 XSS 蠕虫的传播是难以实现的,因此,如何获得真实的实验数据是一个实验难题。为了解决这个难题,本文的实验数据包含了模拟数据,模拟数据的产生基于实际 OSN 应用中热门消息的转播情况。

图 3 为在线社交网络应用中消息被转发的次数随时间变化的分布图,其中曲线 A 为热门消息的转发分布图,曲线 B 为普通消息的转发分布图。图 4 为在线社交网络中 XSS 蠕虫感染用户数量随时间变化的分布图。从图 3 ~4 可看出:1)对比图 3 中的曲线 A 和曲线 B,曲线 A 代表的热门消息在在线社交网络中被转发次数随时间变化的增长率比曲线 B 代表的普通消息更大(曲线 A 比曲线 B 更陡峭),即热门消息在在线社交网络中的传播速度更快;2)对比图 3 和图 4 中的曲线[16],曲线 A 和曲线 C 在增长阶段斜率都较大,与曲线 B 相比,在上升阶段,曲线 A 的分布图更接近于曲线 C,即与普通消息在 OSN 中的传播情况相比,热门消息的传播更近似于 XSS 蠕虫的传播。因此,在模拟实验中,选择普通消息作为数

据集中的良性样本,而热门消息用于模拟传播速度较快的恶意样本。

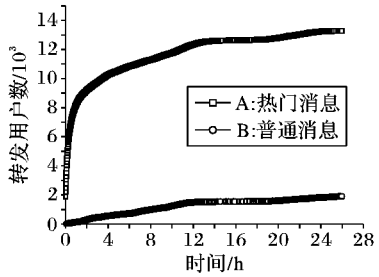

图 3 微博中热门消息和普通消息被转发次数随时间变化的分布

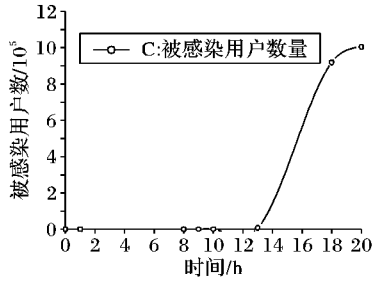

图 4 蠕虫 Samy 感染 MySpace 用户数量随时间变化的分布

5.2 实验数据

实验数据包含了良性样本和恶意样本,其中,良性样本共 28 869 个,来自于 DMOZ 数据库(http://www.dmoz.org);恶意样本包括了由 XSSed 数据库(http:// www.xssed.com)中提供的 13 935 个网页和 2 060 个来自于实际网络应用(http://weibo.com)中的网页,该组网页包含了被多次转发的热门消息(热门消息用于模拟 XSS 蠕虫代码)。

在实验中,选择热门的微博消息(例如,2013 年 6 月 21 日的一条热门微博,http://e.weibo.com/1642292081/zCqR0n4mI)作为 XSS 蠕虫代码的模拟对象,将其在 OSN 中被转发的速度作为 XSS 蠕虫在 OSN 中传播速度的模拟值。计算消息被转发速度时,需要统计该条消息每一时刻在过去的单位时间内被转发的次数,该统计值将作为 XSS 蠕虫代码传播速度的模拟值添加到网页样本的特征向量中。

5.3 结果分析

实验中,利用开源软件 weka 提供的类 weka.classifiers.* 处理训练数据集,以此建立 ADTree 算法的分类器。训练分类器时,选择类中默认的参数值作为训练条件,其中 boosting 的迭代次数为 3。对于改进n-gram模型,通过对特征向量中不同特征值的关系进行分析,实验选择 $n=4$,即网页的改进n-gram模型的形式为(id, $value1$, $value2$, $value3$),其中第一项为每一组的标识符,后三项为特征向量中的 3 个特征值,该 3 个值是特征向量中互相关联的最大集合,因此从效率和有效性角度考虑,实验确定了 $n=4$ 的改进n-gram模型。

在实验中,分别利用方案 1 和方案 2 对训练数据中的网页进行检测,并统计检测结果,如表 2~3 所示。从表 2~3 可看出:两种方案对测试样本的检测都获得了较高的精确度和召回率,结果证明分类器和改进n-gram模型组合的检测方案能够有效地检测出在线社交网络中感染了 XSS 恶意代码的网页。

误报率(False Positive Rate, FPR)和漏报率(False Negative Rate, FNR)是评价检测方案有效性的另一标准。表 4 是一组 FPR 和 FNR 的统计结果,分析后可得出以下结论:1)仅使用分类器情况下的检测效果中,FPR 和 FNR 的值为 5%~8%,良好的检测结果证明了提取的特征和分类器方法对 OSN 中的 XSS 检测的有效性。2)仅利用改进n-gram模型的检测结果中,FNR 的值仅为 1%,即利用改进n-gram模型进行检测时,可检测出 99% 的恶意样本。由此可见,改进n-gram模型对恶意网页有很好的识别性,但是,该方法也产生了误报的情况。3)与分类器方法相比,方案 1 具有较小的 FNR 和较大的 FPR,方案 2 具有较小的 FPR 和较大的 FNR,虽然两个方案未能同时获得较低的 FPR 和 FNR,但是,它们均在某一衡量标准中得到较好的检测结果。经过分析,本文认为当获得较低 FNR 时,由于检测变得更严格,导致 FPR 升高;相反,当 FPR 较低时,由于检测变得相对松弛,FNR 就会升高。正如改进n-gram模型的检测效果所示,较严格的检测条件能准确地检测出恶意的样本,但是可能会导致误报的情况。

另外,对比表 4 中的运行时间可看出:仅使用分类器进行检测的执行时间较短;而由于需要与数据库交互,改进n-gram模型进行检测时花费的时间较多。对于本文提出的两种检测方案,检测方案 1 不需要对所有的待检测实例操作数据库,而检测方案 2 对每一个待检测实例都需要读取数据库,因此检测方案 1 的执行时间少于检测方案 2 的执行时间,效率更高。在实际检测中,可以根据系统对检测程度和检测时间的需求,选择更适合的检测方案。

表 2 检测方案 1 实验结果

实际分类	实验分类结果		精确度/%	召回率/%
	恶意	良性		
恶意	3 356	156	86.7	95.6
良性	514	5 805	97.4	91.9

表 3 检测方案 2 实验结果

实际分类	实验分类结果		精确度/%	召回率/%
	恶意	良性		
恶意	3 308	204	89.0	94.2
良性	409	5 910	96.7	93.5

表 4 两种方案与参考检测方法的实验结果比较

方法	误报率/%	漏报率/%	每个样本的运行时间/s
检测方案 1	8.1	4.4	0.163
检测方案 2	6.5	5.8	0.262
仅分类器	8.1	5.4	0.029
仅改进n-gram	22.3	1.0	0.261

6 结语

本文针对在线社交网络中 XSS 攻击的网络安全问题,提出了一种检测网页是否感染 XSS 恶意代码的方法,方法核心思路基于分类器和改进n-gram模型。首先,根据在线社交网络与传统网络应用的相似点与不同点,提取了一组网页特征;其次,提出了一种改进n-gram模型,模型可利用网页特征对网页进行检测;接着,本文设计了两种结合分类器和改进n-gram模型的实现方案用于检测网页;最后,利用包含模拟感染了 XSS 蠕虫的网页的样本集作为实验数据,通过模拟实验,验证了本文方法的有效性。

参考文献:

[1] EMARKETER. Social media [EB/OL]. [2013-01-24] https://www.emarketer.com/Coverage/SocialMedia.aspx.

[2] Symantec. 诺顿 2012 网络安全报告[R/OL]. [2013-01-30]. ht-

tp://wenku.it168.com/d_000529769.shtml.
[3] HASIB A A. Threats of online social networks[J]. IJCSNS International Journal of Computer Science and Network Security, 2009, 9(11): 288-293.
[4] LIVSHITS V B, CUI W. Spectator: detection and containment of JavaScript worms[C]// ATC 2008: Proceedings of the USENIX 2008 Annual Technical Conference on Annual Technical Conference. Berkeley: USENIX Association, 2008: 335-348.
[5] CAO Y, YEGNESWARAN V, POSSAS P, *et al*. PathCutter: severing the self-propagation path of XSS JavaScript worms in social Web networks[EB/OL]. [2013-10-10]. http://www.dnssec-test-dyn.com/sites/default/files/08_2.pdf.
[6] TER LOUW M, VENKATAKRISHNAN V N. Blueprint: robust prevention of cross-site scripting attacks for existing browsers[C]// Proceedings of the 2009 30th IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2009: 331-346.
[7] LIKARISH P, JUNG E, JO I. Obfuscated malicious JavaScript detection using classification techniques[C]// Proceedings of the 2009 4th International Conference on Malicious and Unwanted Software. Piscataway: IEEE Press, 2009: 47-54.
[8] NUNAN A E, SOUTO E, dos SANTOS E M, *et al*. Automatic classification of cross-site scripting in Web pages using document-based and URL-based features[C]// Proceedings of the 2012 IEEE Symposium on Computers and Communications. Piscataway: IEEE Press, 2012: 702-707.
[9] LI W-J, WANG K, STOLFO S J, *et al*. Fileprints: identifying file types by *n*-gram analysis[C]// IAW 2005: Proceedings of the 6th Annual IEEE Systems, Man, and Cybernetics. Piscataway: IEEE Press, 2005: 64-71.
[10] LANZI A, BALZAROTTI D, KRUEGEL C, *et al*. AccessMiner: using system-centric models for malware protection[C]// Proceedings of the 17th ACM Conference on Computer and Communications Security. New York: ACM Press, 2010: 399-412.
[11] KIM B I, IM C T, JUNG H C. Suspicious malicious Web site detection with strength analysis of a JavaScript obfuscation[J]. International Journal of Advanced Science and Technology, 2011, 26: 19-32.
[12] XU W, ZHANG F, ZHU S. Toward worm detection in online social networks[C]// Proceedings of the 26th Annual Computer Security Applications Conference. New York: ACM Press, 2010: 11-20.
[13] FAGHANI M R, SAIDI H. Malware propagation in online social networks[C]// Proceedings of the 2009 4th International Conference on Malicious and Unwanted Software. Piscataway: IEEE Press, 2009: 8-14.
[14] FAGHANI M R, SAIDI H. Social networks' XSS worms[C]// CSE 2009: Proceedings of the 2009 International Conference on Computational Science and Engineering. Piscataway: IEEE Press, 2009, 4: 1137-1141.
[15] FREUND Y, MASON L. The alternating decision tree learning algorithm[C]// ICML 1999: Proceedings of the 16th International Conference on Machine Learning. San Francisco: Morgan Kaufmann Publishers, 1999: 124-133.
[16] GAMY. I'm popular[EB/OL]. [2013-06-10] http://namb.la/popular/

(上接第 1607 页)

试的方式,即在第二次测试中,对频率组的原子的相位上偏移π/2,然后对两次测试结果进行比较,这样就不会发生误判现象。

尽管该算法具有快速性,但它一个明显的不足就是主要针对一阶多项式相位信号,对于其他信号,比如二阶多项式相位信号、三阶多项式相位信号等,就显得无能为力,如何拓展到其他信号,还待进一步的研究。

参考文献:

[1] TOŠIC I, FROSSARD P. Dictionary learning[J]. IEEE Signal Processing Magazine, 2011, 28(2): 27-38.
[2] MALLAT S, ZHANG Z. Matching pursuits with time-frequency dictionaries[J]. IEEE Transactions on Signal Processing, 1993, 41(12): 3397-3414.
[3] CHEN S S, DONOHO D L, SAUNDERS M A. Atomic decomposition by basis pursuit[J]. SIAM Review, 2001, 43(1): 129-159.
[4] PATI Y C, REZAIIFA R, KRISHNAPRASAD P S. Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition[C]// Proceedings of Conference Record of the 27th Asilomar Conference on Signals, Systems and Computers. Piscataway: IEEE Press, 1993: 40-44.
[5] CHINH L A, MINH N DO. Tree-based orthogonal matching pursuit algorithm for signal reconstruction[C]// Proceedings of the 2006 IEEE International Conference Image Processing. Piscataway: IEEE Press, 2006: 1277-1280.
[6] WANG J, WANG Z, LIU Y. Fast algorithm of sparse signal decomposition based on PSO and LM[J]. Laser & Infrared, 2012, 42(2): 227-230. (王菊, 王朝晖, 刘银. 基于 PSO 和 LM 的信号稀疏分解快速算法[J]. 激光与红外, 2012, 42(2): 227-230.)
[7] WANG Z, HE H, WANG J, *et al*. Fast algorithm for image MP sparse decomposition based on FHT and core dictionary[J]. Journal of the China Railway Society, 2012, 34(9): 51-57. (王在磊, 和红杰, 王建英, 等. 基于核心原子库和 FHT 的图像 MP 稀疏分解快速算法[J]. 铁道学报, 2012, 34(9): 51-57.)
[8] LI Y, YIN Z, WANG J. Fast algorithm for MP sparse decomposition and its application in speech recognition[J]. Computer Engineering and Applications, 2010, 46(1): 122-128. (李雨昕, 尹忠科, 王建英. MP 稀疏分解快速算法及其在语音识别中的应用[J]. 计算机工程与应用, 2010, 46(1): 122-128.)
[9] RUBINSTEIN R, ZIBULEVSKY M, ELAD M. Double sparsity learning sparse dictionaries for sparse signal approximation[J]. IEEE Transactions on Signal Processing, 2010, 58(3): 1553-1564.
[10] FORNASIER M, RAUHUT H. Iterative thresholding algorithms[J]. Applied and Computational Harmonic Analysis, 2008, 25(2): 187-208.
[11] GILBERT A C, IWEN M A, STRAUSS M J. Group testing and sparse signal recovery[C]// Proceedings of the 42nd Asilomar Conference Signals, Systems and Computers. Piscataway: IEEE Press, 2008: 1059-1063.
[12] CHERAGHCHI M, HORMATI A, KARBASI A, *et al*. Compressed sensing with probabilistic measurements: a group testing solution[C]// Allerton 2009: Proceedings of the 47th Annual Allerton Conference on Communication, Control, and Computing. Piscataway: IEEE Press, 2009: 30-35.
[13] DJUROVIC I, SIMEUNOVIC M, DJUKANOVIC S, *et al*. A hybrid CPF-HAF estimation of polynomial-phase signals: detailed statistical analysis[J]. IEEE Transactions on Signal Processing, 2012, 60(10): 5010-5023.
[14] PELEG S, FRIEDLANDER B. The discrete polynomial-phase transform[J]. IEEE Transactions on Signal Processing, 1995, 43(8): 1901-1914.