第 43 卷 第 6A 期
2016 年 6 月
计 算 机 科 学
Computer Science
Vol. 43 No. 6A
June 2016

手机流量非侵入式监测的决策树算法

易军凯 李正东 李 辉
(北京化工大学信息科学与技术学院 北京 100029)

摘 要 针对现有手机中不良软件难以监测和识别的问题,提出并实现了手机流量监测系统,采用非侵入式方法获取手机流量数据,根据特征采用 ID3 算法建立决策树模型,再根据此决策树规则对流量数据进行分类。实验结果表明:该方法对手机流量类型的识别准确率在 92% 以上。

关键词 流量监测,决策树,流量特征,ID3 算法
中图法分类号 TP39 **文献标识码** A

Decision Tree Algorithm in Non-invasive Monitoring Cell Phone Traffic

YI Jun-kai LI Zheng-dong LI Hui
(College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China)

Abstract The purpose of this paper is to solve the problem that it is difficult to monitor and identify the malicious software in the cell phone. Aiming at proposing and realizing a mobile traffic monitoring system, we used non-invasive method to get phone traffic data. A decision tree model was established using the ID3 algorithm for the data, and then the traffic data was classified according to the decision tree rule. The experiment results show that the method can identify the traffic flow generated by the cell phone and the recognition accuracy rate can reach more than 92%.

Keywords Traffic monitoring, Decision tree, Traffic features, ID3 algorithm

1 引言

目前对手机应用的安全性监测大多是侵入式监测,如手机安装监测客户端上传应用特征码到服务器端分析[1],直接对应用安装包分析[2],对手机应用程序提取权限信息进行分析[3]等。侵入式监测有一定的局限性,比如对手机中应用的流量使用情况难以全方位监测,并且必须要在被监测的手机中安装客户端。本文采用非侵入式监测来解决这个问题,分为两个步骤:

(1)在无线网络出口的地方建立一个对流量进行抓包和分析提取的系统。对连接到该无线网络上的手机所产生的进行提取和保存,并对其进行可视化。

(2)对抓取到的数据建立决策树模型,并根据该模型对后续流量进行分类,以判断该流量是否是不良应用产生的。

对于异常流量监测的方式,国内外有较多的研究,分析方法和分类方法也各不相同,主要包括以下几个方面:1)采用 Hoeffding 算法对流量进行分析和分类[4];2)对训练样本动态重建分类模型,然后采用自主学习的方式智能分类[5];3)采用半监督学习方法结合空间分类构建对流量的分类器[6];4)基于向量机(SVM)的数据流分类[7];5)根据数据包的签名进行分类监测[8];6)对信息熵进行评估并使用特征向量分布进行监测[9];7)使用朴素贝叶斯算法进行流量分类[10]。

2 数据提取

2.1 数据提取流程

对手机所连接的无线网络出口处产生的流量数据抓包,并将格式化后的数据入库以进行后续建模和分类操作,对分类后的数据可视化并对异常流量进行警告,具体流程如图 1 所示。

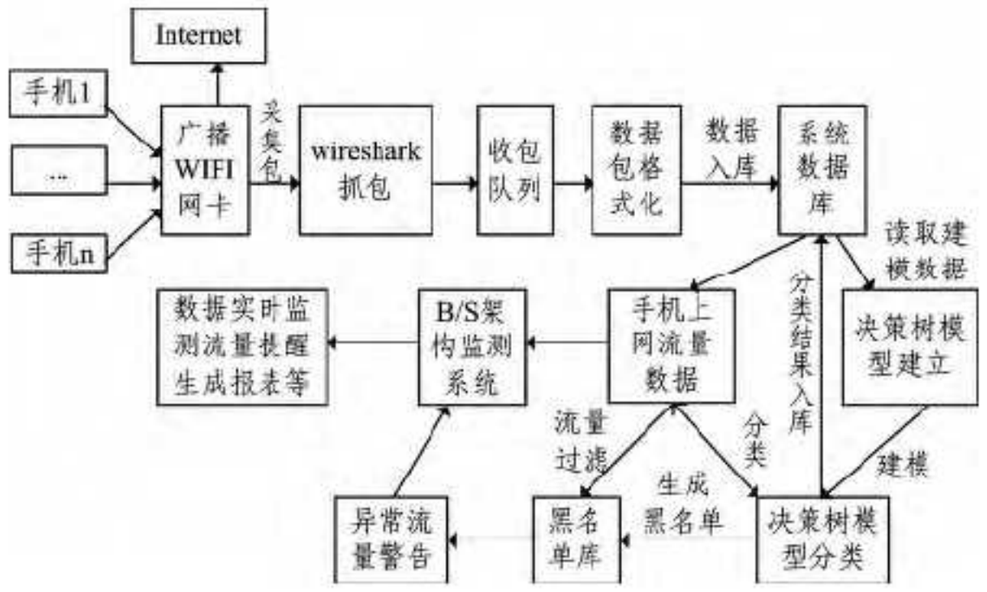

图 1 手机流量分析监测结构图

2.2 数据预处理

本系统在建模前,先将数据库中所存的数据按照建模所用的向量来进行处理,根据请求来源的 MAC 地址和请求的目标 IP 地址进行整理,整理出上行流量总和、下行流量总和、总数据流量、访问的地址所包含的二级链接数等数据,并将数

本文受北京化工大学学科建设项目基金(XK1520)资助。
易军凯(1972—),男,博士,教授,主要研究方向为信息安全技术、优化调度技术,E-mail:yijk@mail.buct.edu.cn;李正东(1990—),男,硕士生,主要研究方向为信息安全技术、软件工程;李 辉(1975—),男,博士,副教授,主要研究方向为计算机安全、计算机网络、图形学。

据整理成数据挖掘可读取的格式。其中数据库中的部分数据如表1所列。

表1 数据库部分样本数据

编号	所访问IP	二级链接	访问次数	上行流量	下行流量
1	1.189.72.178	2	2	767	20440
2	101.226.129.199	20	5	782	23676
3	110.246.47.22	1	2	68	730
4	123.125.80.74	4	2	1140	8979
…	…	…	…	…	…

3 决策树的建立

3.1 选取训练向量

(1)页面二级链接数特征

正常软件和木马软件所请求路径的返回结果存在区别，可以根据二者的不同点进行区分，正常网页浏览的页面中一般也包含多级页面，木马软件访问路径与上传下载相似，二级链接一般没有或较少。

(2)访问时间频次特征

用户通过App访问这些页面时在页面停留的时间较长，而且有很大的概率接着点击访问该页面上的链接。这样，用户访问该地址的时间阈值就较大，而木马软件访问URL的时间阈值则相对较小。

(3)上下行流量特征

手机中正常应用在访问网络时一般下行流量会大于上行流量，木马软件则是由服务器进行控制，在远程控制端发送指定的命令到手机后，手机中的木马软件进行响应处理，将命令对应需要得到的数据进行上传，上行流量一般会大于下行流量。

(4)总数据流量特征

对于恶意广告或者木马应用来说，一般在访问到与之相关的站点后直接进行流量耗费巨大的应用下载任务，其中就可能包含木马应用。

3.2 监测模型的构造

在抓取到的数据中可以发现一个规律，当URL包含的二级链接数较多的时候，访问时间的阈值也较大，单位时间内的访问频率也较高。所以这里对URL包含的二级链接数、上行流量与下行流量比、总数据流量进行分析，如表2所列。

表2 分析的数据项

URL二级链接数	上下行流量比	总数据流量	可能来源
多	小	小	正常APP
多	小	大	正常APP
多	小	大	恶意APP
多	大	大	正常APP
少	小	小	正常APP
少	小	大	恶意APP
少	大	大	木马APP
少	大	小	木马APP

将数据分为两块，$X=\{$URL二级链接数，上下行流量比，总数据流量$\}$，$Y=\{$可能来源$\}$。现阶段的目的是建立一棵决策树，让计算机自动去寻找最合适的映射关系，即$Y=f(X)$，X称为样本，Y称为结果(行为/类)。样本是多维的，$X=\{x_1,x_2,\cdots,x_n\}$，这里$X=\{$URL二级链接数，上下行流量比，总数据流量$\}$，通过这些不同维度的观测记录数据和应对的不同结果找到规律(映射关系)。X的多维不同的数据影响着Y的最终决策。X的多维数据对决策的影响也不相同，优先级高的数据对决策的影响相对较大。影响程度以及对正确结果影响的可信度可以通过训练样本来评判。

通过信息论的熵(Entropy)来衡量样本的可信度。其中用来测量混乱程度的熵的公式为：

$$E(A)=\sum_{j=1}^{v}\frac{s_{ij}+\cdots+s_{mj}}{S}*I(s_{ij},\cdots,s_{mj}),j=1,2,\cdots,v \quad (1)$$

$$I(s_1,\cdots,s_m)=-\sum_{i=1}^{m}p_i*\log_2(p_i),i=1,2,\cdots,m \quad (2)$$

在式(2)中，设S是s个数据样本的集合。假定类标号向量具有m个不同值，定义m个不同的类$C_i(i=1,2,\cdots,m)$。设S_i是类C_i中的样本数。这样式(2)可以给定样本分类所需的期望信息。其中p_i是第i个样本向量属于C_i的概率，并用s_i/s_j估计。

在式(1)中，设向量A具有v个不同的值$\{a_1,a_2,\cdots,a_v\}$，可以用向量A将S划分成v个子集$\{S_1,S_2,\cdots,S_v\}$，其中S_j包含S中这样一些样本：它们在A上具有值a_j。式(2)给出的是根据A划分成子集的熵或期望信息。公式中$\frac{(s_{ij}+\cdots+s_{mj})}{S}$充当第$j$个子集的权，并且等于子集(即$A$值为$a_j$)中的样本个数除以$S$中的样本总数。熵值越小，子集划分的纯度越高。其中，$p_{ij}=\frac{s_{ij}}{|S_j|}$是$S_j$中的样本属于$C_i$类的概率。

当信息一致且所有样本都属于一个类别I时，熵为0；如果样本完全随机，那么熵为1，表明这个特征向量对这种结果状态的预测没有帮助

下面计算上面数据项中的每个维度的数据对结果的影响。通过计算熵来判别。其中每个维度对结果的影响如下：

$E(\text{URL二级链接数多})=-(3/4)*\log(3/4)-(1/4)*\log(1/4)=0.2442$

$E(\text{URL二级链接数少})=-(1/4)*\log(1/4)-(1/4)*\log(1/4)-(2/4)*\log(2/4)=0.4515$

$E(\text{上下行流量比小})=-(3/5)*\log(3/5)-(2/5)*\log(2/5)=0.2922$

$E(\text{上下行流量比大})=-(1/3)*\log(1/3)-(2/3)*\log(2/3)=0.2764$

$E(\text{总数据流量小})=-(3/4)*\log(3/4)-(1/4)*\log(1/4)=0.2442$

$E(\text{总数据流量大})=-(2/5)*\log(2/5)-(2/5)*\log(2/5)-(1/5)*\log(1/5)=0.4581$

得到熵后，每个维度向量的可信度再用信息增益(Information Gain)来衡量。在构造决策树模型的过程中使用贪婪算法，由上到下，递归地构造树。在刚开始构造树时，树根处是所有的训练样本，训练样本的向量是可分类的，如果是连续值，就要提前对其进行离散化；然后根据选择的向量对样本进行递归式的划分。在选择测试向量的时候，参考统计度量值如信息增益值。同样地，在决策树的各个分支节点选择向量时，也要与决策树选择向量一样采用信息增益等方法。当树的某个节点上的样本都属于相同的类，所有的向量都用到了，而且当前没有样本了时，停止划分树。

$Gain(Sample, Action) = E(Sample) - SUM(|Sample(v)|/Sample * E(Sample(v)))$

下面计算各个向量的信息增益：

$Gain(\text{URL 二级链接数}) = E(S) - (4/8) * E(\text{URL 二级链接数多}) - (4/8) * E(\text{URL 二级链接数少}) = 1 - (4/8) * 0.2442 - (4/8) * 0.4515 = 0.6521$

$Gain(\text{上下行流量比}) = E(S) - (5/8) * E(\text{上下行流量比小}) - (3/8) * E(\text{上下行流量比大}) = 1 - (5/8) * 0.2922 - (3/8) * 0.2764 = 0.7137$

$Gain(\text{总数据流量}) = E(S) - (3/8) * E(\text{总数据流量小}) - (5/8) * E(\text{总数据流量大}) = 1 - (5/8) * 0.2442 - (3/8) * 0.4581 = 0.6756$

接着，通过信息增益结果进行决策树的训练，从树根往下依次用最大和最小的结果进行决策树的构造。上面计算出的各个维度的信息增益情况为：上下行流量比>总数据流量>URL 二级链接数。

根据计算的信息增益结果建立的决策树如图 2 所示。

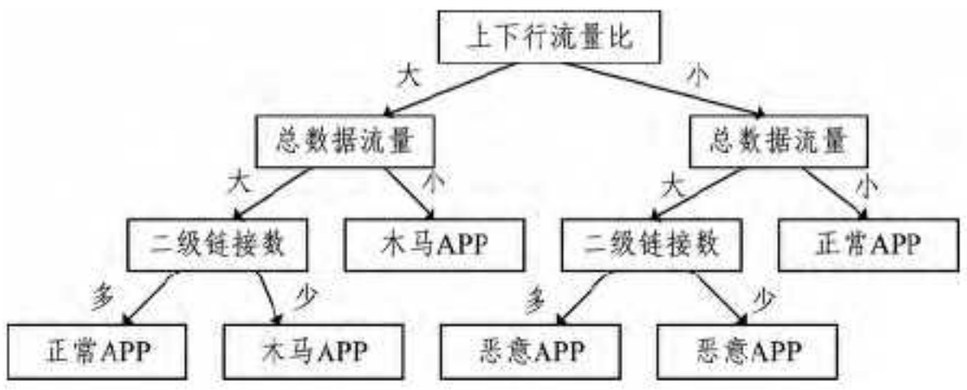

图 2 理论分析的决策树模型

4 实验结果与分析

4.1 样本训练

本实验采用 ID3 算法对 1000 组样本进行训练，建立决策树规则。因版面限制，选取部分样本数据进行展示，如表 4 所列。表中的数据向量和向量类别如表 3 所列。

表 3 数据向量名称和类别

名称	LN	TP	VF	TT	TY
类别	MLN	HTP	HVF	MTT	SF
	FLN	LTP	LVF	FTT	DG
	NLN	MTP			

表 3 中各个向量名称和向量类别的解释如下：

(1)LN 表示网络请求中包含的二级链接数，MLN 表示网络请求中包含的二级链接数较多，本文中将多于 4 个二级链接的作为二级链接数较多的情况。FLN 表示二级链接数较少，本文中将 1 到 4 个二级链接数作为二级链接数较少的情况。NLN 表示无二级链接的情况。

(2)TP 表示上行流量与下行流量的比例，LTP 表示上行流量与下行流量比例低，本文将上下行流量比例在 0.5 以下的作为比例低的情况；将高于该值的作为比例高的情况，表示为 HTP；当下行流量为 0 时，标记为 MTP。

(3)VF 表示单位时间的访问频率，HVF 表示单位时间的访问频率高，本文将每小时访问次数在 4 次及以上的作为高访问频率；次数低于 4 次表示访问频率低，标记为 LVF。

(4)TT 表示总的数据流量，也就是上行流量与下行流量之和。TY 表示流量类型。将流量总量超过训练样本平均流量总量 1.5 倍的情况作为流量多的情况，否则作为流量少的情况。

(5)TY 表示该流量类型，SF 表示正常流量，DG 表示非正常流量。

本文以 WEKA 为工具，对训练样本数据（见表 4）采用 ID3 算法建立决策树规则，进行训练，所得到的决策树模型如图 3 所示。同时也采用了 J48 算法对训练样本数据进行决策树构建的实验，所得到的决策树模型与图 3 所示的相同。

表 4 训练样本数据

样本编号	向量				流量类型
	LN	TP	VF	TT	TY
1	MLN	LTP	HVF	MTT	SF
2	NLN	HTP	LVF	FTT	DG
…	…	…	…	…	…
999	FLN	LTP	LVF	FTT	SF
1000	NLN	HTP	HVF	MTT	DG

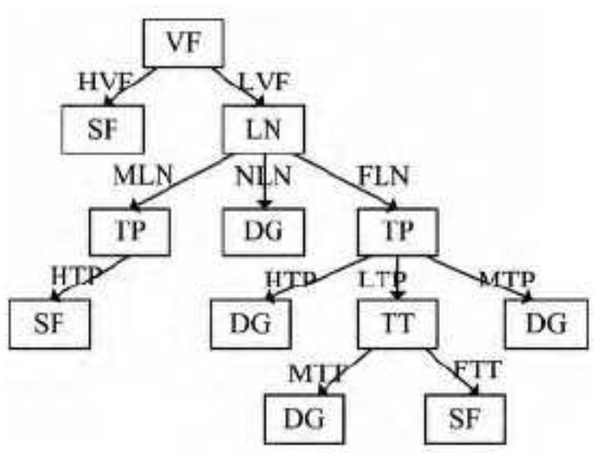

图 3 样本数据的决策树模型图

4.2 流量监测与结果分析

根据上面得到的决策树规则，系统对后续 2595 条实例数据进行分类，同时监测根据决策树分类的准确率。

采用更多不同数量的实例数据进行分类，应用决策树模型对流量进行决策，得到的结果如表 5、表 6 所列。

表 5 ID3 分类效果评价

实例总数	正确分类数量	正确率(%)	平均绝对误差	均方根误差	相对绝对误差(%)
500	470	94	0.0969	0.2232	22.6265
1000	940	94	0.1013	0.2267	23.0163
1500	1396	93.07	0.1139	0.2397	26.6148
2000	1861	93.05	0.1134	0.2385	27.122
2500	2318	92.72	0.1168	0.2421	27.4652

表 6 J48 分类效果评价

实例总数	正确分类数量	正确率(%)	平均绝对误差	均方根误差	相对绝对误差(%)
500	450	90	0.1	0.3162	23.3568
1000	890	89	0.11	0.3317	24.9841
1500	1253	83.53	0.1647	0.4058	38.4817
2000	1678	83.9	0.161	0.4012	38.5094
2500	2074	82.96	0.1704	0.4128	40.055

由监测测试效果可以看出，随着监测规模的扩大，应用决策树模型分类的正确率大体维持在一个较为平稳的水平，ID3 算法决策分类的正确率保持在 92% 以上。由于正常应用产生流量的行为和木马病毒应用产生流量的行为在某些情况下会出现相似的情况，因此进行分类时不可避免地会出现一些误报，以上测试结果显示，错误率可控制在 8% 以内。

图 4 将 ID3 决策树算法与 J48 算法在本系统中的分类结果进行了对比。

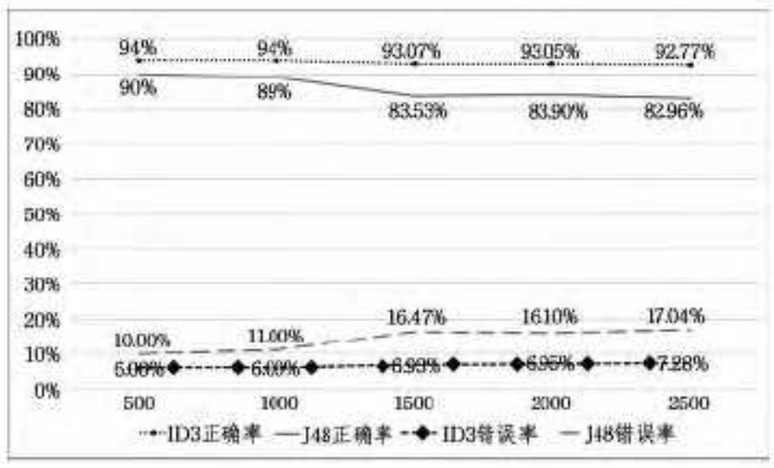

图 4　分类结果正确率与错误率折线图

如图 4 所示，以上算法均能在监测规模扩大时将监测的正确率保持在一个较高的范围内，而 ID3 算法更适用于本系统，准确率更高，同时稳定性也更好。

结束语　针对手机中的应用软件产生的异常流量进行监测十分不便的问题，本文在手机所连接的无线网络出口处使用非侵入式进行流量提取并建立决策树模型进行分类。通过实验结果可以看出，本文所研究的系统对手机异常流量具有较高的监测率，这样为推断手机中的不良软件提供了一种新的方法，可以提高连接在本系统的数台手机的安全性，尤其是在多部设备产生巨大规模流量和数据记录时，本文的研究内容在处理效率和准确率上就更显优势。

参 考 文 献

[1] 文伟平，梅瑞，宁戈，等. Android 恶意软件检测技术分析和应用研究[J]. 通信学报，2014，35(8)：78-85

[2] 杨欢，张玉清，胡予濮，等. 基于多类特征的 Android 应用恶意行为检测系统[J]. 计算机学报，2014，37(01)：15-27

[3] 周裕娟，张红梅，张向利，等. 基于 Android 权限信息的恶意软件检测[J]. 计算机应用研究，2015，32(10)

[4] Carela-Español V，Barlet-Ros P，Bifet A，et al. A streaming flow-based technique for traffic classification applied to 12＋1 years of Internet traffic[J]. Telecommunication Systems，2015：1-14

[5] Divakaran D M，Su L，Liau Y S，et al. SLIC：Self-Learning Intelligent Classifier for Network Traffic[J]. Computer Networks，2015，91：283-297

[6] Chen Z，Liu Z，Peng L，et al. A novel semi-supervised learning method for Internet application identification[J]. Soft Computing，2015：1-13

[7] Groleat T，Arzel M，Vaton S. Stretching the Edges of SVM Traffic Classification With FPGA Acceleration[J]. IEEE Transactions on Network & Service Management，2014，11(3)：278-291

[8] Tegeler F，Fu X，Vigna G，et al. BotFinder：finding bots in network traffic without deep packet inspection[C]// Proc Co-next. 2012：349-360

[9] Wang B，Chua K C，Srinivasan V，et al. Information Coverage in Randomly Deployed Wireless Sensor Networks[J]. IEEE Transactions on Wireless Communications，2007，6(8)：2994-3004

[10] Narayanan V A，Sureshkumar V，Rajeswari A. Automatic Traffic Classification Using Machine Learning Algorithm for Policy-Based Routing in UMTS-WLAN Interworking[M]// Artificial Intelligence and Evolutionary Algorithms in Engineering Systems. Springer India，2015：305-312

(上接第 331 页)

立在实验的基础上的，在实际应用中需要进行改进和完善。

结束语　通过对 Web 应用跨站脚本漏洞进行分析，提出了结合模糊测试和遗传算法的 Web 应用跨站脚本漏洞挖掘方法，以优化 XSS 漏洞特征测试用例的生成方法及动态挖掘流程。在该方法中，通过定义漏洞库模糊测试方法、过滤规则和补全规则进行 XSS 漏洞特征选择与提取，并采用遗传变异规则来搜索特征空间，对 XSS 特征测试用例进行过滤、补全、选择、交叉、变异操作，通过多次反复迭代选出最优的 XSS 攻击特征测试用例，由此实现 Web 应用跨站脚本漏洞的挖掘。分析表明，该方法能有效挖掘 Web 应用中跨站脚本漏洞。

但是，本文目前针对 XSS 漏洞库的构建主要是人工输入，且 XSS 漏洞遗传变异只考虑了一维变异，故如何实现 XSS 漏洞库的自动输入及如何构建 XSS 漏洞特征测试用例的多维变异[18]是下一步的工作重点。

参 考 文 献

[1] 中国互联网络信息中心(CNNIC). 第 27 次中国互联网络发展状况统计报告(2015-2)[R/OL]. http://www.cnnic.cn/hlwfzyj/hlwxzbg/201502/P020150203551802054676.pdf

[2] OWASP. OWASP Top Ten Project[R]. 2013

[3] Stuttard D，Pinto M，石华耀. 黑客攻防技术宝典：Web 实战篇[M]. 2009

[4] 刘为. 基于模糊测试的 XSS 漏洞检测系统研究与实现[D]. 长沙：湖南大学，2010

[5] 温凯，郭帆，余敏. 自适应的 Web 攻击异常检测方法[J]. 计算机应用，2012，32(7)：2003-2006

[6] 吴子敬，张宪忠，管磊，等. 基于反过滤规则集和自动爬虫的 XSS 漏洞深度挖掘技术[J]. 北京理工大学学报，2012，32(4)：396-396

[7] 潘古兵，周彦晖. 基于静态分析和动态检测的 XSS 漏洞发现[J]. 计算机科学，2012，39(B6)：51-53

[8] 吴少华，程书宝，胡勇. 基于 SVM 的 Web 攻击检测技术[J]. 计算机科学，2015，42(S1)：362-364

[9] Shahriar H，Zulkernine M. S2XS2：a server side approach to automatically detect XSS attacks[C]// 2011 Ninth IEEE International Conference on Dependable，Autonomic and Secure Computing. Sydney：IEEE Press，2011：7-14

[10] 王春东，邱晓华. 基于特征策略的 XSS 漏洞检测技术研究[J]. 天津理工大学学报，2013，29(5)：25-29

[11] 许静，练坤梅，田伟，等. 应用遗传算法自动生成 XSS 跨站点脚本漏洞检测参数的方法[P]. 中国，2015-05-30

[12] 朱红萍，巩青歌，雷战波. 基于遗传算法的入侵检测特征选择[J]. 计算机应用研究，2012，29(4)：1417-1419

[13] 郭慧，王晓菊，刘明艳，等. 基于遗传算法的入侵检测系统特征选择方法研究[J]. 华北科技学院学报，2014，11(9)：68-72

[14] 韦存堂. SQL 注入与 XSS 攻击自动化检测关键技术研究[D]. 北京：北京邮电大学，2015

[15] 牛皓. 基于网络爬虫的 XSS 漏洞检测系统的研究与设计[D]. 北京：北京邮电大学，2015

[16] https://www.owasp.org/index.php/XSS—Filter—Evasion—Cheat—Sheet

[17] 潘古兵. Web 应用程序渗透测试方法研究[D]. 重庆：西南大学，2012

[18] 吴志勇，王红川，孙乐昌，等. 遗传算法在多维 Fuzzing 技术中的应用[J]. 小型微型计算机系统，2011，32(5)：998-1004

[19] 王云. 基于爬虫和模糊测试的 XSS 漏洞检测工具设计与实现[D]. 广州：华南理工大学，2015