第49卷第2期 中国科学技术大学学报 Vol. 49, No. 2
2019年2月 JOURNAL OF UNIVERSITY OF SCIENCE AND TECHNOLOGY OF CHINA Feb. 2019

文章编号:0253-2778(2019)02-0166-07

基于贝叶斯网络的XSS攻击检测方法

王培超,周 鋆,朱 承,张维明

(国防科技大学信息系统工程重点实验室,湖南长沙 410073)

摘要:跨站脚本(XSS)攻击是最严重的网络攻击之一.传统的XSS检测方法主要从漏洞本身入手,多依赖于静态分析和动态分析,在多样化的攻击载荷(payload)面前显得力不从心.为此提出一种基于贝叶斯网络的XSS攻击检测方法,通过领域知识获取该网络中的节点.利用领域知识构建的本体为贝叶斯网络的构建提供良好的特征选择基础,并从中提取了17个特征,同时从公开渠道搜集的恶意IP和恶意域名为该模型及时检测新型攻击补充有力规则.为验证所提方法的有效性,在实际收集的XSS攻击数据集上进行实验,结果表明,在面对多样化的攻击时,该方法可以保持90%以上的检测准确率.

关键词:跨站脚本(XSS)攻击检测;贝叶斯网络;领域知识;恶意IP;恶意域名

中图分类号:TP309 **文献标识码**:A doi:10.3969/j.issn.0253-2778.2019.02.012

引用格式:王培超,周鋆,朱承,等. 基于贝叶斯网络的XSS攻击检测方法[J]. 中国科学技术大学学报,2019,49(2):166-172.

WANG Peichao,ZHOU Yun, ZHU Cheng,et al. XSS attack detection based on Bayesian network[J]. Journal of University of Science and Technology of China, 2019,49(2):166-172.

XSS attack detection based on Bayesian network

WANG Peichao,ZHOU Yun, ZHU Cheng, ZHANG Weiming

(*Science and Technology on Information Systems Engineering Laboratory*, *National University of Defense Technology*, *Changsha*, 410073, *China*)

Abstract: Cross-site scripting (XSS) attack is one of the most serious cyber-attacks. Traditional XSS detection methods mainly focus on the vulnerability itself, relying on static analysis and dynamic analysis, which appear weak in defending the flood of various kinds of payloads. An XSS attack detection method is proposed based on the Bayesian network, in which the nodes are acquired with domain knowledge. The ontology constructed with domain knowledge provides a good basis for feature selection, and 17 features have been abstracted from it; besides, malicious IPs and malicious domain names collected from open source channels make effective complement rules for the detection of new attacks. To validate the proposed method, experiments were conducted on a collected real-world dataset about XSS attacks. The results show that the proposed method could maintain a detection accuracy of above 90%.

Key words: cross-site scripting (XSS) attack detection; bayesian network; domain knowledge; malicious IP; malicious domain name

收稿日期:2018-10-04;**修回日期**:2018-12-04

基金项目:国家自然科学基金(61703416),湖南省自然科学基金(2018JJ3614)资助.

作者简介:王培超,男1993年生,硕士生.研究方向:人工智能. E-mail:peichaow@163.com

通讯作者:周鋆,博士/讲师. E-mail:zhouyun@nudt.edu.cn

0　引言

跨站脚本(cross-site scription, XSS)攻击是一种危害巨大的 Web 攻击,其对个人隐私和社会经济已经造成了巨大的损害[1]. XSS 漏洞是一种非常常见的 Web 漏洞;当存在这种漏洞时,攻击者将恶意代码注入 Web 页面或是请求数据中,当受害者浏览相应页面或被引诱点击了包含恶意代码的链接后,就会遭受 XSS 攻击. XSS 经常与其他 Web 攻击组合使用,造成用户隐私信息和用户会话的失窃,对 Web 安全具有巨大威胁.

XSS 攻击通常可以分成两种:非持久型 XSS(non-persistent XSS)和持久型 XSS(persistent XSS). 非持久型 XSS 是一种一次性的攻击,攻击者通常将恶意代码写在 URL 请求中并诱骗受害者点击相应的恶意 URL,当用户访问该链接时,就会遭受 XSS 攻击. 持久型 XSS 中,恶意代码会被攻击者提交到数据库中,在数据库中持久存在,任何使用了包含该恶意数据的页面的用户都有可能遭受攻击。

为了防范 XSS 攻击,目前主要有两种方法被广泛应用:静态分析和动态分析[2],这两种方法的目的是及时发现 XSS 漏洞:①静态分析是在获取源代码的条件下,在不运行程序时进行的分析,这种方法通常需要大量的领域知识;②动态分析是在程序运行时对程序可能存在的漏洞进行测试,这种方法依赖于攻击向量的完整性,会造成较高的误报率. 在攻击的检测方面,运用特定规则是主流方法,数据驱动的分析方法是一种新兴的方法,多个安全产品中已经开始将这一方法应用于实践. 机器学习[3]为检测 XSS 攻击提供了有效的数据驱动的方法,然而少有模型会考虑利用开源威胁情报对模型效果进行改进(例如使用恶意 IP 库来提升检测准确率). 这些威胁情报包含了从多个渠道获得的信息,通常会有最新类型攻击的描述[4].

本文提出了一种基于贝叶斯网络(Bayesian network,BN)的 XSS 攻击检测方法,将 XSS 攻击的检测问题建模为一个使用贝叶斯网络[5]模型的二分类问题. 为构建贝叶斯网络模型,本文首先从构建出的 XSS 攻击本体中提取知识来构建特征,之后运用贝叶斯网络结构学习算法对网络的结构和相应参数进行学习,并利用从公开渠道获取的恶意域名和恶意 IP 对模型的准确性进行改善. 通过对在 GitHub 以及各大安全博客论坛中收集整理得到的实际数据集上进行实验,结果验证了本文所提方法的有效性.

1　相关研究现状

在检测 XSS 攻击方面,传统方法主要从漏洞本身出发,有两种方法:静态分析和动态分析[6].

静态分析是一种常用的手工分析方法,包括词法分析和数据流分析等,网络安全人员直接从代码中挖掘可能存在的漏洞;动态分析是另一种重要的 XSS 漏洞检测方式,测试者在不同的输入点上尝试各种类型的攻击载荷(payload),模仿现实中攻击者对应用程序进行渗透测试. 传统方法的研究已经取得了很多成果,如 Agosta 等[7]制作了一款将结合静态污点分析与特征码分析的 XSS 漏洞检测工具;Nenad 等[8]开发了一种基于数据流分析的 XSS 漏洞检测工具对 PHP 应用进行检测;王丹等[9]提出了一种基于隐马尔科夫模型的动态攻击向量生成方法;Fabien 等[10]提出了可以自行生成恶意脚本并对其进行评估的 XSS 模糊测试工具 KameleonFuzz.

传统的漏洞检测方法在面对多样化的攻击载荷时其效果难以令人满意,且大量的人工参与、攻击向量的完整性等问题会对结果产生重要影响. 对于可能的攻击进行及时检测和拦截是当前防范 XSS 的有效方法;近年来,机器学习算法被引入 XSS 攻击检测[11-12]并取得了较好的结果. 贝叶斯网络是机器学习领域中一个常用的白盒模型(white-box model),其在网络安全领域已经有了越来越多的应用[13-14]. 贝叶斯网络应用中,模型的构建通常需要结合专家经验和高质数据. 贝叶斯网络学习算法是解决贝叶斯网络构建的重要方法,包含参数学习和结构学习两部分. 参数学习中,网络结构是已知的,根据已有数据获取变量的条件概率表(conditional probability table,CPT);结构学习中,贝叶斯网络的参数和结构均未知,两者都需要从现有数据中学习获取. 结构学习算法主要分为 3 种:基于约束的算法、基于评分搜索的算法和混合算法. 大量的高质数据[15]是纯数据驱动的贝叶斯结构学习所必需的,而高质数据的获取一直是一个难题,因而领域知识(如安全本体[16])和规则的支持在贝叶斯网络构建中必不可少.

近期的研究引入开源威胁情报(open source intelligence),并将其作为模型构建的补充[17]. 威胁情报是基于证据的知识,可以为决策提供大量的信息支持[18]. 通用漏洞评分系统(common

vulnerability scoring system,CVSS)是对漏洞进行定量化评估的标准,已被多次使用[19-20]. 除了漏洞库之外,还有诸如恶意 IP 库和恶意域名库等开源的威胁情报没有被利用. 本文利用这部分开源威胁情报,将恶意 IP 库和恶意域名库作为补充规则加入基础贝叶斯网络模型,构建统一的 XSS 攻击检测框架,以便对新型攻击进行及时的检测.

2 模型构建

2.1 模型应用过程

图 1 阐述了用本文方法进行 XSS 检测的过程. 该方法首先构建本体对 XSS 攻击进行高层建模,并从中提取可以反映 XSS 攻击的特征. 然后将这些特征作为贝叶斯网络的节点并从原始数据中获取相应节点的值作为贝叶斯网络的训练数据,利用基于评分搜索的算法对贝叶斯网络的结构和参数进行获取. 在基本的模型构建好后,对恶意域名和恶意 IP 信息进行搜集和爬取,这些开源威胁情报作为模型的补充规则改善对其检测效果. 在实际应用中,当待判断数据被输入模型时,该数据会经由基本模型和补充规则的双重判断. 贝叶斯网络的白盒模型的特点意味着其结果对用户的良好可解释性,而从恶意域名和恶意 IP 衍生而来的补充规则可以及时发现隐藏的攻击类型,模型和规则的结合显著提高了对攻击的检测能力,可以有效帮助安全人员对事件进行处理.

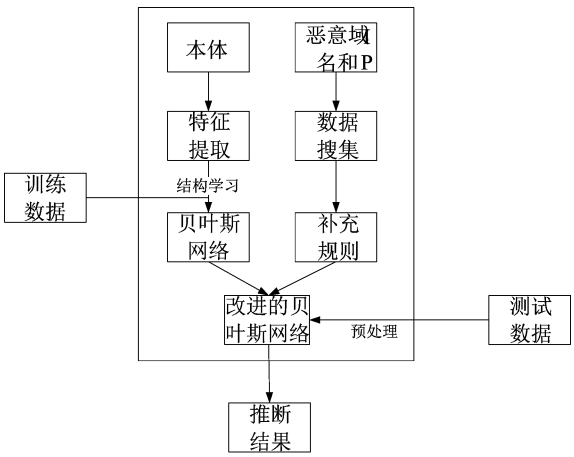

图 1 模型应用过程框架

Fig. 1 Overall framework of applying the proposed method

2.2 基于领域知识的模型构建

本体是领域知识的重要表示形式,在计算机领域中本体的经典概念由 Gruber[21] 给出. 本体是实体间概念和关系的形式化描述[22],提供了一个特定领域知识的概况. 在本文中,本体为后续的建模提供了知识库,有助于在后续建模中把握 XSS 攻击的核心概念和特点. 直接用本体建模语言,对本体进行构造是一个耗时巨大的工作,因而在实际的构造过程中通常会利用图形化工具辅助工作. 图形化工具可以让本体的构造者专注于理顺实体间概念和关系,不需要掌握底层的编写语言. 本文使用 Protégé 作为本体建模工具,该工具是一个免费的开源本体构建框架,使用 OWL(web ontology language)本体建模语言,可以帮助研究者对本体进行高效地构建. 本文构建的 XSS 攻击的高层本体如图 2 所示.

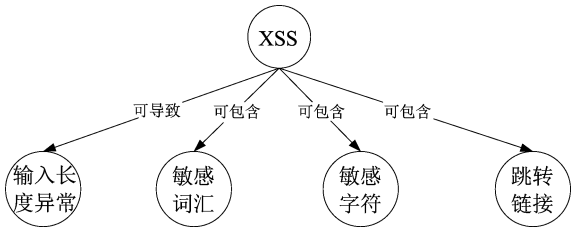

图 2 XSS 高层本体

Fig. 2 High-level ontology of XSS

本体构建好后,便可以从这些高层概念中提取可以表示 XSS 攻击的特征用于后续贝叶斯网络模型的构建,每个特征在贝叶斯网络中都以节点的形式出现. 贝叶斯网络是一个有向无环图,其节点代表随机变量,节点间的箭线代表随机变量间的依赖关系[15]. 令 $X=\{X_1,\cdots,X_n\}$ 代表一个随机变量(连续或离散)的集合,集合中每个变量 X_i 都取有限个值或在一定的范围内取值. 贝叶斯网络可以用一个二元组(S,P)进行形式化的表示,其中 S 代表贝叶斯网络的结构,该结构蕴含了存在于 X 中变量间的关系,P 是与每个变量相关的条件概率分布的集合. 进一步地,$P=\{P_i\}$,$P_i=P(X_i \mid Pa_i)$,Pa_i 代表节点集合 X 中 X_i 的父节点,节点满足马尔科夫性质的联合概率分布,且

$$P(X)=\prod_i P(X_i \mid Pa_i) \tag{1}$$

对于一个节点代表离散变量的贝叶斯网络来说,每一个节点的概率分布情况由条件概率表(conditional probability table,CPT)给出,相应节点的 CPT 包含了在给定父节点的取值的情况下当前节点取特定值的概率大小[23]. 在获取了特征和原始数据后,便可以利用贝叶斯结构学习算法构建模型. 本文使用基于评分搜索的算法进行贝叶斯结构学习,在该类型的算法中,评分函数用于评估当前结构和数据的适合度,而搜索算法用于在候选集合中选

取具有最高得分的结构. 具体来说,本文选用 BDeu (Bayesian Dirichlet equivalent uniform)[24] 作为评分函数,使用禁忌搜索(Tabu search)[25] 作为搜索算法.

禁忌搜索是在最优化问题中寻找最优解的一个常见搜索算法. 为了避免局部最优,该算法在搜索过程中应用禁忌列表(Tabu list)来防止对最优值的重复计算,使该算法更易得到全局最优解而非局部最优解. BDeu 评分由 BDe 和 BDu 发展而来,是在假设所有可能的有向无环图(directed acyclic graph, DAG)符合均匀先验分布时,在给定数据的情况下最大化取当前有向无环图的概率.

2.3　利用恶意 IP 和恶意域名信息改进模型

实际应用中,单纯依赖机器学习模型进行检测而忽视威胁情报会为攻击者的攻击埋下隐患. 网络威胁情报(cyber threat intelligence,CTI)是对传统网络安全的重要补充,英国国家基础设施保护中心(the UK's Centre for the Protection of National in Frastructure,CPNI)将威胁情报划分为 4 类①:战术(tactical)、技术(technical)、行动(operational)和策略(strategic). 这种分类可以有效地帮助研究人员对威胁情报进行宏观把握,然而这种威胁情报在现实中多半被营利性组织所掌控,属于团体的内部资产,非相应团体成员无法获得. 对于研究人员来说,其可利用的威胁情报主要从开源网站上获取,本文在 XSS 攻击的检测过程中利用的是恶意 IP 地址库和恶意域名库.

恶意 IP 地址库是研究人员最容易获得的威胁情报之一,这些威胁情报被用于浏览器的访问黑名单中,当客户端发出的请求中包含这些地址时,浏览器便会进行阻拦. 有很多公开网站发布恶意 IP 列表,其中,FireHOL② 维护的恶意 IP 库是一个非常好的选择,该数据库融合了多个来源的恶意 IP 数据,并且更新频率较高,是研究者获取恶意 IP 数据的重要来源. 恶意域名库主要用于防范对僵尸网络、钓鱼页面和恶意软件. 当一个用户访问的页面地址被包含在恶意域名列表中时,为用户安全考虑会对用户告警. 很多网站都有向大众公开的相应的恶意域名库(例如 DNS-BH project③ 和 PhishTank④),研究中可以从这些公开网站上获取相应数据.

开源威胁情报可以作为补充规则对模型效果进行改进. 令 $C=\{C_1,\cdots,C_m\}$ 为补充规则的集合,$X=\{X_1,\cdots,X_n\}$ 为依据本体提取出的特征的集合,$R=\{R_1,\cdots,R_N\}$ 为数据记录的集合,$r=\{r_1,\cdots,r_N\}$ 为依据特定规则提取出的值,$T=\{T_1,\cdots,T_N\}$ 为相应记录的预测结果. T_i 的计算公式为

$$T_i=\begin{cases}1, & \text{if } \max\{I_C(r_i),P(T_i=1\mid X_1,\cdots,X_n)\}\geqslant\mu\\ 0, & \text{if } \max\{I_C(r_i),P(T_i=1\mid X_1,\cdots,X_n)\}<\mu\end{cases}\tag{2}$$

式中,0 代表当前记录为正常,1 代表当前记录为 XSS 攻击,μ 取值为 0.5; $I_C(r_i)$ 是一个指示函数,当依据特定规则提取的值被补充规则检测到时该指示函数的值为 1,否则为 0; $P(T_i=1\mid X_1,\cdots,X_n)$ 表示在给定相应证据后通过联结树算法(junction tree algorithm)计算得到的后验概率,意味着贝叶斯网络标签节点的马尔科夫边界(Markov blanket)中节点的取值. 如果 $I_C(r_i)$ 的值为 1,则该条记录的检测结果为 XSS 攻击;若 $I_C(r_i)$ 值为 0 并且 $P(T_i=1\mid X_1,\cdots,X_n)\geqslant\mu$,意味着 $P(T_i=1\mid X_1,\cdots,X_n)\geqslant P(T_i=0\mid X_1,\cdots,X_n)$,那么当前的记录仍被检测为 XSS 攻击;若 $I_C(r_i)$ 值为 0 并且 $P(T_i=1\mid X_1,\cdots,X_n)<\mu$,意味着 $P(T_i=1\mid X_1,\cdots,X_n)<P(T_i=0\mid X_1,\cdots,X_n)$,那么当前的记录被检测为正常. 获得检测结果后,可以利用诊断推理对结果进行进一步的分析来获取相应的原因. 同时,如果输入数据因相应的补充规则而被检测为攻击,相应的原因同样很容易获取.

3　实验和讨论

3.1　实验准备

本文实验使用的主要工具是 Python 3.6 和 SPSS 24.0,训练数据集主要从 GitHub⑤ 上获取. 该数据集具有 151 658 条记录,包含了 16 151 条 XSS 攻击载荷和 135 507 条正常的请求数据. 三种类型的 XSS 攻击其在触发机制、存储位置和输出位置上均不尽相同,然而其攻击载荷有很多相似之处,例如都包含有敏感字符和敏感词汇. 测试集中的 XSS 攻击载荷是从 GitHub 和多个传播网络安全知识的博客等地方收集的,该测试集共包含 10 000 条数据,

① https://en.wikipedia.org/wiki/Cyber_threat_intelligence
② https://github.com/firehol/blocklist-ipsets
③ http://www.malwaredomains.com/
④ https://www.phishtank.com
⑤ https://github.com/duoergun0729/1book/tree/master/data

其中正常数据 6 503 条，XSS 攻击载荷 3 497 条，这些正常的数据是从训练集中随机采样得到的.

根据之前构建好的本体提取 XSS 攻击特征. 本文实验针对非持久型 XSS 这一常见的 XSS 攻击形式进行特征的提取，包含 XSS 代码的请求通常比正常请求的长度要长，因而将输入长度作为一个特征；恶意脚本包含敏感关键词和敏感字符，XSS 代码主要是由 JavaScript 编写的，因此那些可以改变原始请求的语义并执行新的脚本的字符和关键词应该被关注. 仅考虑输入长度、字符和关键词是不够的，为了增加迷惑性，攻击者会采用重定向技术，在一个请求中放入跳转链接并将真正的恶意代码隐藏在跳转后的页面中，因此需要利用开源威胁情报对这种恶意的跳转进行检测. 本文总共根据构建好的本体提取出 18 个特征，将这些特征 No. 0 开始编号一直到 No. 17，其中 No. 17 代表该条记录的类型(正常还是 XSS 攻击). 这些特征的细节如表 1 所示.

表 1 贝叶斯网络中使用的特征

Tab. 1 Features used in the BN model

序号	含义	序号	含义	序号	含义
0	输入长度	6	eval 数量	12	双引号(")数量
1	alert 数量	7	src 数量	13	左尖括号(<)数量
2	script 数量	8	prompt 数量	14	右尖括号(>)数量
3	onerror 数量	9	href 数量	15	反斜杠(\\)数量
4	img 数量	10	javascript 数量	16	逗号(,)数量
5	onload 数量	11	单引号(')数量		

3.2 模型学习

上小节已经将所需特征进行了提取，本文将这些特征作为贝叶斯网络中的节点. 由于一个攻击者可以通过对原始输入数据进行编码或改变大小写等方式对原始请求进行隐蔽，因此本文首先对数据进行预处理，包含如下步骤：URL 解码、JS 解码、HTML 实体编码的解码、ASCII 码的解码，将解码后的数据中大写字母全部转变为小写字母. 为在建模过程中降低计算的复杂度，这里使用基于熵的离散化方法对初步预处理后的数据集进行离散化. 基于熵的离散化是一种有监督的离散化方法，本文使用 SPSS 24. 0 来完成这一任务.

预处理完毕后，本文用 Python 扩展包 PyAgrum[①] 进行结构学习. PyAgrum 扩展包包含很多成熟的贝叶斯学习算法，覆盖参数学习、结构学习和推断. 这里采用其内建的禁忌搜索和 BDeu 评分进行结构学习，学习得到的贝叶斯网络结构如图 3 所示.

除了数据驱动的贝叶斯网络学习，如果请求数据中包含跳转，需要将跳转目的地(IP 形式或域名形式)取出，并利用开源威胁情报对这些跳转目的地进行检测. 本实验采用的恶意 IP 库和恶意域名库包含 3 个来源：FireHOL、DNS-BH project 和 PhishTank，其中 FireHOL 用于提供恶意 IP 库，后两者用于提供恶意域名库. 本实验利用 Python 3. 6

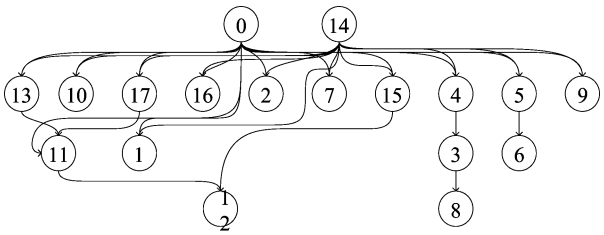

图 3 结构学习获取的贝叶斯网络结构图

Fig. 3 The BN structure learnt by structure learning algorithm

的扩展包 Requests 库[②]和 BeautifulSoup 库[③]进行以上数据的爬取解析，Requests 库是 Python 语言编写的爬虫库，可以方便地从网上爬取大量的数据；BeautifulSoup 库是一个可以从 HTML 或 XML 文件中提取数据的 Python 库，能够有效解析爬取下来的数据并根据用户需求对其提供所需数据.

在包含有恶意跳转的 XSS 载荷中，如果使用 IP 来标记跳转地址，则该 IP 地址通常会进行伪装. IP 地址通常用点分十进制的表示方式(如 127. 0. 0. 1)，每个部分是 0～255 之前的十进制整数，其每一部分都可以用二进制、八进制、十进制或十六进制表示. IP

① http://agrum.gitlab.io/

② http://www.python-requests.org/en/master/

③ https://www.crummy.com/software/BeautifulSoup/bs4/doc/index.zh.html

地址的 4 个部分间不需要统一的进制形式，通过进制混用可以对恶意 IP 地址进行伪装. 同时，IP 地址也可以用单一的整数进行表示. 将跳转的链接取出后，若为 IP 地址则将其还原为点分十进制的形式后再与爬取得到的开源威胁情报进行比对，若为域名标志的跳转则与恶意域名库进行比对，若匹配成功则该条记录为恶意的攻击载荷. 当经过比对后没有发现该链接为恶意链接时，根据构建好的贝叶斯网络的特征对该条记录进行相应特征值的提取，并将这些值作为证据输入贝叶斯网络，根据标签节点中不同标签的概率来判断该条记录是否为恶意的攻击载荷.

3.3　模型效果

为了比较本文方法与传统分类器的效果，选用以下常见分类器进行对比实验：SVM、朴素贝叶斯（naïve Bayes，NB）、逻辑斯蒂（logistics regression，LR）和决策树（decision tree，DT），这些分类器已经包含在 scikit-learn① 这一 Python 的机器学习扩展包中. 为代表利用跳转实施攻击的 XSS，将爬取的恶意 IP 与恶意域名随机替换到存在跳转的 XSS 样本或常规请求，显然这样产生的所有样本都是恶意样本，其示例如表 2 所示.

表 2　正常请求与恶意跳转请求的样例

Tab. 2　Samples of normal and malicious redirection requests

正常样本	替换后的恶意样本
＜A HREF＝"http://220.181.57.216/""＞Click＜/A＞	＜A HREF＝"http://66.000146.0×7.147/""＞Click＜/A＞
＜SCRIPT SRC＝ http://www.qq.com/＞＜/SCRIPT＞	＜SCRIPT SRC＝http:// leonarderickson.chez.com/xss.js＞＜/SCRIPT＞
＜iframe src＝http://www.baidu.com＞	＜iframe src＝http://ccjbox.ivyro.net/index.html＞

将这些样本按照不同比例随机替换到测试集中，比较本文方法及传统分类器，其分类准确性结果如表 3 所示.

表 3　不同方法实验结果对比

Tab. 3　Experimental results of different methods

替换比例	本文方法	SVM	LR	NB	DT
0%	91.04%	97.00%	98.02%	99.18%	98.04%
5%	91.57%	94.65%	95.59%	95.46%	94.36%
10%	92.07%	92.26%	93.18%	91.77%	90.66%
15%	92.56%	89.86%	90.71%	88.07%	87.01%
20%	92.88%	87.49%	88.28%	84.38%	83.32%
25%	93.35%	85.07%	85.87%	80.67%	79.64%
30%	93.85%	82.73%	83.40%	77.01%	75.95%
35%	94.23%	80.32%	80.98%	73.27%	72.27%
40%	94.67%	77.94%	78.55%	69.60%	68.58%
45%	95.08%	75.53%	76.09%	65.89%	64.86%

从表 3 可以看出，虽然在替换比例较低的情况下本文方法效果不是最佳，但是随着替换比例的升高，传统分类器效果下降，本文方法效果稳步上升，尤其是当替换比例达到 15%之后. 当测试集替换比例达到最大时，本文方法仍可以达到 95.08%的准确度，此时其他方法的准确度已经低于 80%. 本文方法优于其他方法的原因很明显，转换后的恶意地址可以避免诸如敏感词和字符之类的伪装，并且转换后的 IP 地址可以使有效载荷比正常有效载荷短，而借助于整合的威胁情报，本文方法可以及时发现这些隐藏的攻击.

4　结论

本文提出了一种基于贝叶斯网络的 XSS 攻击检测方法，通过从本体中获取的知识提取 XSS 攻击的特征，并使用结构学习来获得贝叶斯网络的结构，同时实验中收集的恶意 IP 和恶意域名信息为本文方法提供了效果的改进.

参考文献（References）

[1] FONSECA J, SEIXAS N, VIEIRA M, et al. Analysis of field data on web security vulnerabilities[J]. IEEE Transactions on Dependable & Secure Computing, 2014, 11(2): 89-100.

[2] GUO X B, JIN S Y, ZHANG Y. X XSS vulnerability detection using optimized attack vector repertory[C]// International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery. Xi'an, China: IEEE, 2015: 29-36.

[3] 戴桦，李景，卢新岱，等. 智能检测 WebShell 的机器学习算法[J]. 网络与信息安全学报，2017，3(4)：51-57.
DAIH, LI J, LU D X, et al. Machine learning algorithm for intelligent detection of WebShell[J]. Chinese Journal of Network & Information Security, 2017, 3(4): 51-57.

① http://scikit-learn.org/stable/

[4] BURGER E W, GOODMAN M D, KAMPANAKIS P, et al. Taxonomy model for cyber threat intelligence information exchange technologies[C]// Workshop on Information Sharing & Collaborative Security. Scottsdale, USA: ACM, 2014: 51-60.

[5] ZHOU Y, FENTON N, NEIL M. Bayesian network approach to multinomial parameter learning using data and expert judgments[J]. International Journal of Approximate Reasoning, 2014, 55(5): 1252-1268.

[6] VOGT P, NENTWICH F, JOVANOVIC N, et al. Cross site scripting prevention with dynamic data tainting and static analysis[C]// Proceedings of 14th Annual Network & Distributed System Security Symposium. San Diego, USA: ACM, 2007: 74-83.

[7] AGOSTA G, BARENGHI A, PARATA A, et al. Automated security analysis of dynamic web applications through symbolic code execution[C]// 9th International Conference on Information Technology: New Generations. Las Vegas, USA: IEEE, 2012: 189-194.

[8] JOVANOVIC N, KRUEGEL C, KIRDA E. Pixy: A static analysis tool for detecting web application vulnerabilities[C]// IEEE Symposium on Security and Privacy. Berkeley, USA: IEEE, 2006: 258-263.

[9] 王丹, 顾明昌, 赵文兵. 跨站脚本漏洞渗透测试技术[J]. 哈尔滨工程大学学报, 2017, 38(11): 1769-1774.
WANG D, GU M C, ZHAO WB. Cross-site script vulnerability penetration testing technology[J]. Journal of Harbin Engineering University, 2017, 38(11): 1769-1774.

[10] DUCHENE F, RAWAT S, RICHIER J L, et al. KameleonFuzz: Evolutionary fuzzing for black-box XSS detection[C]// 4th ACM Conference on Data and Application Security and Privacy. San Antonio, USA: ACM, 2014: 37-48.

[11] KRISHNAVENI S, SATHIYAKUMARI K. Multiclass classification of XSS web page attack using machine learning techniques[J]. International Journal of Computer Applications, 2013, 74(12): 36-40.

[12] AHMED M A, ALI F. Multiple-path testing for cross site scripting using genetic algorithms[J]. Journal of Systems Architecture, 2016, 64: 50-62.

[13] WU J Y, YIN L H, GUO Y C. Cyber attacks prediction model based on Bayesian network[C]// 18th International Conference on Parallel and Distributed Systems. Singapore: IEEE, 2012: 730-731.

[14] AXELRAD E T, STICHA P J, BRDICZKA O, et al. A Bayesian network model for predicting insider threats[C]// Proceedings of the Security and Privacy Workshops. San Francisco: IEEE, 2013: 82-89.

[15] ZHOU Y, FENTON N, ZHU C. An empirical study of Bayesian network parameter learning with monotonic influence constraints[J]. Decision Support Systems, 2016, 87: 69-79.

[16] FENZ S. An ontology-based approach for constructing Bayesian networks [J]. Data & Knowledge Engineering, 2012, 73(2): 73-88.

[17] CAGLAYAN A, TOOTHAKER M, DRAPEAU D, et al. Behavioral analysis of botnets for threat intelligence[J]. Information Systems and e-Business Management, 2012, 10(4): 491-519.

[18] TOUNSI W, RAIS H. A survey on technical threat intelligence in the age of sophisticated cyber attacks [J]. Computers & Security, 2017, 72: 212-233.

[19] MUñOZ-GONZáLEZ L, SGANDURRA D, PAUDICE A, et al. Efficient attack graph analysis through approximate inference[J]. ACM Transactions on Privacy and Security, 2017, 1(1): 1-31.

[20] XIE P, LI J H, OU X, et al. Using Bayesian networks for cyber security analysis [C]// International Conference on Dependable Systems & Networks. Chicago, USA: IEEE, 2010, 1: 211-220.

[21] GRUBER T R. A translation approach to portable ontology specifications[J]. Knowledge Acquisition, 1993, 5(2): 199-220.

[22] GRUBER T R. Toward principles for the design of ontologies used for knowledge sharing [J]. International Journal of Human-Computer Studies, 1995, 43(5-6): 907-928.

[23] ZHOU Y, FENTON N, HOSPEDALES T M, et al. Probabilistic graphical models parameter learning with transferred prior and constraints[EB/OL]. [2018-05-18], Uncertainty in Artificial Intelligence, http://auai.org/uai2015/proceedings/papers/190.pdf.

[24] HECKERMAN D, DAN G, CHICKERING D M. Learning Bayesian networks: The combination of knowledge and statistical data [J]. Uncertainty Proceedings, 1994, 20(3): 293-301.

[25] GLOVER F. Tabu search: A tutorial[J]. Interfaces, 1990, 20(4): 74-94.