基于网络行为模糊模式识别的蠕虫检测方法

严 芬[1] 陈霜霜[1,2] 殷新春[1]

(扬州大学信息工程学院 扬州 225127)[1] (盐城师范学院信息科学与技术学院 盐城 224002)[2]

摘 要 网络蠕虫攻击由于危害大、攻击范围广、传播速度快而成为因特网危害最大的攻击方式之一。如何有效地检测网络蠕虫攻击是当前网络安全研究领域的一个重要方向。通过对网络蠕虫攻击行为的分析和研究,提出了一种根据蠕虫爆发时产生的典型网络行为来检测未知蠕虫的方法。该算法通过分别学习正常主机和受感染主机的网络行为建立相应的标准分类模糊子集,然后利用模糊模式识别法判定待测主机是否感染蠕虫。最后进行实验验证,结果表明,该方法对未知扫描类蠕虫有较好的检测效果。

关键词 蠕虫,检测,模糊模式识别,网络行为

中图法分类号 TP393 **文献标识码** A

Worm Detection Method Based on Fuzzy Pattern Recognition to Network Behaviors

YAN Fen[1] CHEN Shuang-shuang[1,2] YIN Xin-chun[1]

(College of Information Engineering, Yangzhou University, Yangzhou 225127, China)[1]

(College of Information Science and Technology, Yancheng Teachers University, Yancheng 224002, China)[2]

Abstract Worms have been one of the most serious threats to Internet security due to the significant damage, large range of victims and fast spread. How to detect network worm attack is an important aspect of network security research area. This article proposed a method which detects worms by analyzing and studying typical network behaviors while worms burst out. The algorithm studies the network behaviors of normal and abnormal computer separately, establishes standard fuzzy subsets of classification, and judges if the observation computer infects worms by utilizing fuzzy pattern recognition method. Finally, the experiment with the worm applications in the real world proves that this method is able to detect unknown scanning worms preferably.

Keywords Worm, Detection, Fuzzy pattern recognition, Network behavior

1 引言

网络蠕虫是一种能够自我复制、攻击能力强、恶意的计算机程序或代码,通过网络传播对计算机和网络造成严重破坏。近年来,网络蠕虫利用快速的传播速度、多样化的传播途径,给计算机网络以及人们的日常生活造成了越来越大的影响。蠕虫通过对主机进行漏洞扫描来发现新的目标,从而进行快速传播。其中,顺序扫描和随机扫描的蠕虫最多[1]。与正常的网络操作相比,蠕虫在扫描时具有以下特点:①会向目标主机发送大量的连接请求数据包,以判断这些目标主机是否存在、是否开机、是否提供相应的服务,目标主机的系统或应用软件是否存在漏洞,以及是否可以被感染等[2]。在这些连接请求数据包中,绝大多数为"第一次连接"[3]。在实际网络环境中,攻击目标的选择具有盲目性,信息的搜集和探测都会导致大量IP地址的产生,其中80%左右为无效IP地址。②感染蠕虫的主机的时间间隔会很短,而正常主机在进行网络访问的时候,对外发起连接请求的时间间隔会比较长。因此,蠕虫扫描会引起网络中新连接速度过快、失败连接数过多、第一次失败连接数很高等一些典型的网络行为[4]。

根据蠕虫扫描时所产生的以上典型网络行为,本文引用了"第一次连接"(FCC, First Contact Connection)的概念,将FCC分为4种情形,并建立了关于FCC 4种情形的论域,利用模糊模式识别法来判定待测主机是否感染了蠕虫。

2 相关工作

基于网络行为的蠕虫检测技术是当前研究的热点。文献[5]提出了通过判断失败连接数是否达到阈值来检测远程蠕虫的方法。该方法中,阈值的选择是比较困难的,过低或过高都会影响检测效果,而且对于不同蠕虫而言,阈值的选取结果可能不一样。文献[6]提出了一种利用连续假设检验来检测蠕虫扫描的TRW检测方法。该方法基于蠕虫扫描时将产生较高的FCC失败率的网络行为,通过考察主机每个连接状态是失败还是成功,采用连续假设检验的方法对失败连接次数和成功连接次数做比较,如果失败连接次数与成功连接次数的差值超过了判断阈值,则判定该主机感染蠕虫。该方法虽然较文献[5]有了改进,但由于仅基于FCC失败概率高这一

到稿日期:2012-08-30 返修日期:2012-11-22 本文受江苏省高校自然科学研究项目(10KJB520020),国家863计划项目(2007AA01Z448),江苏省科技支撑计划基金项目(BE2008124)资助。

严 芬(1978-),女,博士,副教授,主要研究方向为网络与信息安全,E-mail:yanfen@yzu.edu.cn;陈霜霜(1983-),女,硕士生,助教,主要研究方向为网络与信息安全;殷新春(1962-),男,博士,教授,主要研究方向为密码学与信息安全。

网络行为进行检测，因此对于 P2P 及游戏等应用程序仍存在较高的误报率。故还需要进一步挖掘其他网络行为作为蠕虫检测指标。文献[1]中提出了 FP-RS 检测方法，该方法新增了连接请求检测指标。当某台主机的 FCC 数量达到给定阈值 NCE 时，将对该主机进行检测。在判断主机 FCC 失败概率的基础上，通过对主机 FCC 请求大小的重尾特性的估计，判断该主机是否感染蠕虫。此方法的局限性在于如何合理确定阈值 NCE，且检测结果对连接请求大小的分布存在依赖，对于“带宽延迟型”蠕虫可能会逃避该检测技术。此外，文献[1]还提出了 FP-AI 检测技术，该技术在判断主机 FCC 失败概率的基础上，增加了连接到达时间间隔的检测指标，采用贝叶斯方法计算其后验概率，通过比较后验概率是否超过设定阈值来判断主机是否感染蠕虫。这两种检测技术在不增加漏报的情况下，均很大程度地降低了误报，其原因在于采用了除 FCC 失败概率之外的检测指标。

本文在参考 FP-AI 检测技术的基础上，为了降低漏报率和误报率，采用了 FCC 失败连接概率和 FCC 连接速度两个检测指标作为检测依据，提出了基于网络行为模糊模式识别的蠕虫检测方法。该方法只需经过简单的数值计算和模糊处理即可以有效地检测变速、慢速等未知扫描类蠕虫，从而能够避免上述方法中阈值取值困难的问题，而且对 P2P 应用程序不易产生误报。

3 检测原理及算法

3.1 定义

定义 1(第一次连接，FCC) 考察一台主机，若以该主机的 IP 地址为源地址，FCC 为在某个考察期内向之前从未连接过的目标 IP 地址发起请求的连接[2]。

定义 2(模糊子集) 给定论域 U 上的一个模糊子集 A，就是给定论域 U 到区间[0,1]的一个映射，且有：

$\mu_A: U \to [0,1]$

$u \mapsto \mu_A(u) \in [0,1]$

映射 μ_A 叫做模糊子集 A 的隶属函数，$u \in U$ 对应着一个确定值 $\mu_A(u) \in [0,1]$，$\mu_A(u)$ 叫做 $u \in U$ 对 A 的隶属度[7]。

定义 3(模糊幂集) 论域 U 上全体模糊子集所构成的一个集合，称为 U 的模糊幂集，记为 $F(U)$，即 $F(U)=\{A \mid \mu_A: U \to [0,1]\}$[7]。

定义 4(贴近度) 设对于 $A,B \in F(U)$，有对应的实数 $q(A,B)$，且满足：

①$0 \leqslant q(A,B) \leqslant 1$

②$q(A,B)=q(B,A)$

③当 $A \subseteq B \subseteq C$ 时，$q(A,C) \leqslant q(A,B) \wedge q(B,C)$

则称 q 为 $F(U)$ 中的贴近度，称 $q(A,B)$ 为模糊集 A 与 B 的贴近度[7]。

定义 5(择近原则) 设 U 为论域，q 是 F 中的贴近度。$A_1, A_2, \cdots, A_n$ 是 U 的 n 个模糊子集，A^* 是 U 的一个模糊子集，即待识别对象。如果满足 $q(A^*, A_j)=\max\limits_{i \leqslant n} q(A^*, A_i)$，其中，$j=1,2,\cdots n$，则待识别对象 A^* 与模糊模式 A_j 最贴近，即认为 A^* 相对属于 A_j[7]。

3.2 检测原理

根据主机当前产生的 FCC 连接是成功还是失败，以及本次 FCC 连接与上次 FCC 连接之间的时间间隔是大于还是小于设定的时间间隔，将该主机发送的连接请求分为 X_1、X_2、X_3、X_4 4 种情况。X_1 为 $Itv > Interval$ 且连接失败，X_2 为 $Itv > Interval$ 且连接成功，X_3 为 $Itv < Interval$ 且连接失败，X_4 为 $Itv < Interval$ 且连接成功。其中，Itv 表示本次 FCC 连接与上次 FCC 连接间的时间间隔，$Interval$ 表示设定的时间间隔，取值为 0.2s[1]。将这 4 种连接请求情况定义为一个论域 $U=\{X_1, X_2, X_3, X_4\}$，定义该论域 U 上的一个模糊子集 $Q=\{\mu_1/X_1, \mu_2/X_2, \mu_3/X_3, \mu_4/X_4\}$，表示该主机发起的连接请求行为 X_i 隶属于该模糊集的程度是 μ_i，其中 $i=1\sim4$，$0 \leqslant \mu_i \leqslant 1$。正常主机、待测主机以及感染蠕虫的主机都可用模糊子集 Q 描述。

设某时间段内正常主机中某连接请求行为 X_i 出现的频率均值为 $E(A_i)$，而其在感染蠕虫主机中出现的频率均值为 $E(B_i)$，从统计学角度分析，正态分布隶属函数更能体现人的直觉推理方式[8]，且偏大方的模糊现象用偏大型隶属函数表示较合理，所以选用 F 分布中的正态偏大型模糊分布来构造正常主机的连接请求行为集 m 的隶属函数[9]：

$$\mu_m(E(A_i))=1-e^{-(\frac{E(A_i)}{\sigma})^2}, E(A_i) \geqslant 0 \quad (1)$$

式中，$i=1\sim4$，$\sigma=\max\{E(A_1), E(A_2), E(A_3), E(A_4)\}/4$。

同样，构造感染蠕虫主机的连接请求行为集 n 的隶属函数：

$$\mu_n(E(B_i))=1-e^{-(\frac{E(B_i)}{\sigma})^2}, E(B_i) \geqslant 0 \quad (2)$$

式中，$i=1\sim4$，$\sigma=\max\{E(B_1), E(B_2), E(B_3), E(B_4)\}/4$。

对于待测主机，构造其连接请求行为集 v 的隶属函数：

$$\mu_v(A_i)=1-e^{-(\frac{A_i}{\sigma})^2}, A_i \geqslant 0$$

式中，$i=1\sim4$，$\sigma=\max\{A_1, A_2, A_3, A_4\}/4$。

在自学习阶段，对正常主机训练数据集中每个连接请求行为 X_i 出现的频率进行统计，得出频率均值，并根据隶属函数 $\mu_m(E(A_i))$ 计算得到正常标准模糊子集：$m=\{\mu_1^m/X_1, \mu_2^m/X_2, \mu_3^m/X_3, \mu_4^m/X_4\}$，对感染蠕虫的主机训练数据集做同样的处理，可以得到异常标准模糊子集：$n=\{\mu_1^n/X_1, \mu_2^n/X_2, \mu_3^n/X_3, \mu_4^n/X_4\}$。

检测时，在滑动时间窗口内对待测主机的网络连接进行监测，当攻击发生时，统计待测主机各连接行为出现的频率，根据其隶属函数进行计算可得到模糊集：$v=\{\mu_1^v/X_1, \mu_2^v/X_2, \mu_3^v/X_3, \mu_4^v/X_4\}$。然后，计算 m 与 v 之间的贴近度 $q(m,v)$ 以及 n 与 v 之间的贴近度 $q(n,v)$。模糊模式识别中有海明贴近度、格贴近度、欧氏贴近度、最大最小贴近度以及算术平均最小值贴近度 5 种计算公式，实验中我们对这 5 种贴近度计算方法分别进行了计算分析和比较，发现算术平均最小贴近度算法更适用于本算法，其计算公式为[10]：

$$\delta(A,B)=\frac{2\sum\limits_{i=1}^{n}\min[\mu_A(u_i), \mu_B(u_i)]}{\sum\limits_{i=1}^{n}[\mu_A(u_i)+\mu_B(u_i)]} \quad (3)$$

计算出模糊集间的贴近度以后，再根据“择近原则”判断待测主机属于正常类还是已感染蠕虫类。

4 算法实现

算法采用的模糊子集的计算过程如下：

(1)分析某时间段内待测主机发送的网络数据包,解析网络数据包中的 TCP 连接请求情况。对于每个连接,解析其相应的目标地址和端口号(Dest,Port)、连接发起时间以及连接状态 *flag*1 和 *flag*2,其中 *flag*1 用于表示 FCC 失败与否(*flag*1=1 表示 FCC 成功,*flag*1=0 表示 FCC 失败),*flag*2 用于表示 *Itv* 是否大于 *Interval*(*flag*2=1 表示 *Itv*>*Interval*,*flag*2=0 表示 *Itv*<*Interval*);检查(Dest,Port)是否在历史目标地址和端口列表中,若在,则说明该连接并非 FCC,无需做进一步处理;否则,将(Dest,Port)添至历史列表中,并记录 *flag*1 和 *flag*2 的值。

(2)根据该时间段内网络数据包的解析结果,计算模糊子集。依据 *flag*1 和 *flag*2 的值对 X_1、X_2、X_3、X_4 4 种情形分别进行统计,计算出这 4 种情形各自出现的频率,再结合相应的隶属函数,求得模糊子集为 $Q=\{\mu_1/X_1,\mu_2/X_2,\mu_3/X_3,\mu_4/X_4\}$。

整个检测过程主要包含了数据采集、样本训练、模糊子集计算、贴近度计算、结果判断等部分。基于第 3.2 节的检测原理,检测过程可以分成以下步骤:

(1)对多个正常样本数据和蠕虫样本数据进行训练,经分析计算后,最终得出正常标准模糊子集 m 和异常标准模糊子集 n,供检测使用;

(2)运行待检测蠕虫样本程序,采集并分析网络数据包,计算出待检测蠕虫样本的模糊子集 v;

(3)依据训练过程得到的正常标准模糊子集 m 和异常标准模糊子集 n,利用算术平均最小贴近度计算公式来计算贴近度 $q(m,v)$和 $q(n,v)$。

(4)根据"择近原则"判断待测主机是属于正常类,还是已感染蠕虫类。

5 实验过程及结果分析

5.1 产生数据集

实验采用 windump 软件在真实网络环境中产生并捕获数据包形成正常和异常数据集。具体过程如下:①正常标准数据集:当计算机处于正常状态时进行多次采样,每次连续采样 20 分钟,为了使采样的数据集具有代表性和科学性,主要考虑以下两方面的因素对计算机性能的影响[11]:a)用户的操作行为。进行数据包采样时打开若干应用程序,如 office、IE 浏览器、Windows Media Player 等;b)后台程序。运行后台程序主要影响处理器时间、页缺失、硬盘读写等系统参数,常见的后台程序如 winRAR。②异常标准数据集:在蠕虫样本选取时,我们主要考虑的因素是蠕虫的不同 IP 扫描策略。同时,通过实验计算分析发现:当异常样本的数量大于 5 时,所计算出的异常标准模糊子集的隶属度非常接近,所以选取了 Dabber. a、Deborm. y、Sasser. a、Lovesan. u、Welchia. s 等 5 种蠕虫程序作为异常样本。分别注入以上 5 种蠕虫样本,每次连续采样 20 分钟,在采样的同时综合考虑用户行为以及后台程序这两个因素。

5.2 实验结果及分析

分别计算采集的正常和异常标准数据集中每个样本各连接请求行为 $X_i(i=1\sim4)$的频率,计算结果分别如表 1 和表 2 所列。

表 1 正常标准数据集中每个样本各连接请求行为的频率

样本 \ 频率	X_1 的频率	X_2 的频率	X_3 的频率	X_4 的频率
正常样本 1	0.353562	0.641161	0.005277	0.000000
正常样本 2	0.278512	0.692562	0.009917	0.019008
正常样本 3	0.267951	0.693520	0.024518	0.014011
正常样本 4	0.655172	0.339080	0.005747	0.000000
正常样本 5	0.324963	0.604136	0.026588	0.044313

表 2 异常标准数据集中每个样本各连接请求行为的频率

样本 \ 频率	X_1 的频率	X_2 的频率	X_3 的频率	X_4 的频率
Dabber. a	0.000000	0.000000	0.101664	0.898336
Deborm. y	0.102002	0.296473	0.071497	0.530029
Sasser. a	0.084932	0.024658	0.008219	0.882192
Lovesan. u	0.000000	0.000000	0.098182	0.901818
Welchia. s	0.007828	0.015656	0.185910	0.790607

根据表 1 和表 2 中的数据进行计算,得出正常标准数据集中各连接请求行为 X_i 频率的均值:$E(A_1)=0.376032$,$E(A_2)=0.594092$,$E(A_3)=0.014409$,$E(A_4)=0.015466$;异常标准数据集中各连接请求行为频率的均值:$E(B_1)=0.038952$,$E(B_2)=0.067357$,$E(B_3)=0.093094$,$E(B_4)=0.800596$。然后,根据式(1)、式(2)计算得出正常和异常标准数据集的模糊子集分别为:

$$m=\{0.998355\}/X_1,0.999999/X_2,0.009368/X_3,0.010785/X_4\}$$

$$n=\{0.037167/X_1,0.107078/X_2,0.194540/X_3,0.999999/X_4\}$$

利用标准数据集进行检测时,首先对 CodeGreen. a、Doomran、Sluter、Dabber. a、Raleka. ab、Padobot. af、Sasser. a 等未知扫描类蠕虫进行了检测。此外,考虑到目前 P2P 程序会产生类蠕虫行为,算法可能产生误报,我们对 PPlive(P2P 视频软件,用于在线直播和点播)和电驴(P2P 下载工具,用于下载资料和影视)这 2 个正常 P2P 程序的数据包也进行了检测。计算待测样本的贴近度时,使用海明贴近度、格贴近度、欧氏贴近度、最大最小贴近度和算术平均最小值贴近度这 5 种贴近度计算方法分别进行了计算。结果表明,式(3)所示的算术平均最小贴近度算法更适合本文,其计算结果如表 3 所列。

表 3 待测样本检测结果

程序名称 \ 结果	与正常标准数据集的贴近度	与异常标准数据集的贴近度	是否误报	是否漏报	是否为蠕虫
CodeGreen. a	0.238939	0.779179	否	否	是
Doomran	0.012178	0.908332	否	否	是
Sluter	0.013386	0.914472	否	否	是
Dabber. a	0.012581	0.939185	否	否	是
Raleka. ab	0.240827	0.845191	否	否	是
Padobot. af	0.251747	0.761096	否	否	是
Sasser. a	0.098981	0.844016	否	否	是
PPLive	0.990971	0.107041	否	否	否
电驴	0.718955	0.590790	否	否	否

从表 3 可以看出,本文提出的检测算法不仅能有效地检测未知扫描类蠕虫,而且对 P2P 程序不易产生误报,具有较低的误报率。实验过程和结果进一步地验证了本文算法的有效性。

结束语 模糊模式识别方法由于能够得到样本属于各个

(下转第 115 页)

during operational outages[J]. Theory and evidence from the UK payment system,2010(2):314-323

[3] Fecht F, Nyborg K G, Rocholl J. The price of liquidity: The effects of market conditions and bank characteristics[J]. Journal of Financial Economics,2011(11):344-362

[4] Borio,Bergh V D. The nature and management of payment system risks:an international perspective[N]. BIS papers,1993

[5] McAndrews J J,Wasilyew G. Simulations of failure in a payment system[A]//Leinonen H,ed. Liquidity,Risks and Speed in Payment and Settlement Systems a Simulation Approach[C]. Finland:Bank of Finland Studies,2005:230-247

[6] Johnson K,McAndrews J J,Soramöki K. Economising liquidity with deferred settlement mechanisms[A]//Leinonen H,ed. Liquidity,Risks and Speed in Payment and Settlement Systems a Simulation Approach[C]. Finland: Bank of Finland Studies, 2005:180-215

[7] Ledrut E. How can banks control their exposure to a failing participant? [A]//Leinonen H,ed. Simulation Studies of Liquidity Needs,Risks and Efficiency in Payment Networks [C]. Finland: Scientific monographs,2007:228-252

[8] Schmitz S,Puhr C. Structure and stability in payment networks-A panel data analysis of ARTIS simulations[A]//Leinonen H, ed. Simulation Analyses and Stress Testing of Payment Networks [C]. Finland:Scientific monographs,2009:146-185

[9] Galbiati M,Soramöki K. An agent-based model of payment systems[A]//Leinonen H,ed. Simulation analyses and stress testing of payment networks [C]. Finland: Scientific monographs, 2009:316-336

[10] Galbiati M,Soramöki K. Liquidity-saving mechanisms and bank behavior[N]. Bank of England:Working Paper No. 400,2010

[11] 贺培. 支付系统监督:开放条件下维系金融安全的重要途径[J]. 国际金融研究,2005(11):67-72

[12] 马征. 支付系统风险分析与控制[J]. 济南金融,2005(9):51-52

[13] 郑建,何秀华. 大额支付系统运行中风险分析及防范途径[J]. 金融实务,2005(12):55-56

[14] 柴小卉,靳力华. 加强我国现代化支付系统风险管理的思考[J]. 金融研究,2006(3):138-145

[15] 付泽宇. 中国现代化支付清算系统的发展及思考[J]. 金融与经济,2006(2):56-59

[16] 吴华茵. 支付系统制度建设相关问题探讨[J]. 金融会计,2006(9):24-25

[17] 吴华茵. 支付系统流动性风险管理探讨[J]. 金融会计,2007(6):33-34

[18] 尹奕文. 银行业流动性风险生成机制研究[J]. 金融在线,2009:32-34

[19] 欧阳卫民. 完善支付环境建设,推动支付系统发展[J]. 金融电子化,2008(11):15-18

[20] 欧阳卫民. 我国支付清算系统的特点和发展趋势[J]. 财经科学,2009(2):34-40

[21] 吉春娴. 支付系统变革对银行流动性管理影响的实证研究[J]. 金融实务,2009(5):54-57

[22] 崔瑜. 我国大额实时支付系统流动性风险控制措施探析[J]. 深圳金融,2011(11)

[23] 邓创,刘晓彤. 关于我国现代化支付结算体系建设的思考——美国支付结算体系的经验与启示[J]. 现代经济,2008(7):4-6

[24] 牛晨,魏先华,潘松,等. 我国大额支付系统中的流动性风险[J]. 系统工程,2010(11):23-28

(上接第 110 页)

类别的不确定程度,建立样本对类别的不确定性描述,因而能更加客观地反映现实世界[12]。本文依据蠕虫扫描时会产生某些典型网络行为的特点,提出了一种基于模糊模式识别的扫描类蠕虫检测算法。利用该方法检测扫描类蠕虫时,只需要进行简单的数值运算和模糊模式的处理,这不仅大大降低了计算的开销,而且能够有效地检测未知的扫描类蠕虫,不会对 P2P 程序产生误报。隶属函数的选取是模糊模式识别方法的重点,目前,隶属函数的选取主要都是依赖于专家经验。当然,本文所提检测算法不能检测拓扑类及被动等待型蠕虫,如 Email 蠕虫、IM 蠕虫,对这类蠕虫的检测还需作进一步地研究与探讨。

参 考 文 献

[1] 陈宇峰. 蠕虫模拟方法和检测技术研究[D]. 杭州:浙江大学,2006

[2] 汪伟. 网络蠕虫检测技术研究与实现[D]. 杭州:浙江大学,2006

[3] Chen Yu-feng,Dong Ya-bo,Lu Dong-ming,et al. Detecting randomly scanning worms based on heavy-tailed property[C]// Proc. of 2005 IEEE International Conference on Networking, Sensing and Control. Orlando,Florida,2005:354-358

[4] 肖枫涛. 基于网络行为的蠕虫检测关键技术研究[D]. 长沙:国防科技大学,2009

[5] Robertson S,Siegel E V,Miller M,et al. Surveillance detection in high bandwidth environments[C]//Proc. of DARPA DISCEX 111 Conefrence. 2003:130-139

[6] Jung J,Paxson V,Begrer A W,et al. Fast Portscan detection using sequential hypothesis testing[C]//Proc. of the IEEE Symposium on Security and Privacy. 2004:211-225

[7] 付文,魏博,赵荣彩,等. 基于模糊推理的程序恶意性分析模型研究[J]. 通信学报,2010,31(1):44-50

[8] 刘向杰,夏靖波,柴天佑. 一类基于正态分布隶属函数的模糊控制策略研究[J]. 控制与决策,1998,13(4):365-368

[9] 顾雨婕. 用于行为分析反木马的模糊分类算法研究[D]. 杭州:浙江工业大学,2008

[10] 李鸿吉. 模糊数学基础及实用算法[M]. 北京:科学出版社. 2005

[11] Moskovitch R,Elovici Y,Rokach L. Detection of unknown computer worms based on behavioral classification of the host[J]. Computational Statistics & Data Analysis,2008,52(9):4544-4566

[12] 李晶皎,赵丽红,王爱侠. 模式识别[M]. 北京:北京电子工业出版社,2010

[13] 况晓辉,黄敏桓,许飞. 网络蠕虫实验环境构建技术研究[J]. 计算机科学,2010,37(7):54-56,73