Internet 蠕虫防范技术研究与进展*

周　涛, 戴冠中, 慕德俊
(西北工业大学 自动化学院, 陕西 西安 710072)

摘　要: 当前蠕虫的频频爆发使得蠕虫问题已成为网络安全领域的焦点问题。分析了蠕虫的特征行为, 研究了国内外几种最新的 Internet 蠕虫防范系统, 并在此基础上展望了蠕虫攻防的发展趋势。

关键词: 蠕虫; 网络安全; 传播模型; 过滤
中图法分类号: TP309　　文献标识码: A　　文章编号: 1001-3695(2006)06-0013-03

Research and Development on Internet Worm Containment Technology

ZHOU Tao, DAI Guan-zhong, MU De-jun
(College of Automatic Control, Northwestern Polytechnical University, Xi'an Shanxi 710072, China)

Abstract: Owing to the frequent explosions of Internet worms, Internet worm has been the focus of attention in cyberspace security field recently. This paper first analyzes Internet worm's characteristics, then describes some Internet worm containment systems which are proposed at home and abroad, and prospects the trend of worm and anti-worm technology in the end.

Key words: Internet Worm; Cyberspace Security; Propagation Model; Filter

1　引言

随着信息化的发展, Internet 已经渗入到人们日常生活的方方面面, Internet 蠕虫造成的安全威胁也日益严重。Internet 蠕虫是指能够独立运行, 并能通过寻找和攻击远方主机的漏洞进行自主传播的恶意代码。自 1988 年 11 月第一个互联网蠕虫莫里斯(Morris)出现以来, 蠕虫就向人们展示了它不同于传统计算机病毒的特点和巨大破坏力。进入 21 世纪后, 蠕虫更是频频大规模爆发, 其爆发频率、扩散速度、造成的危害超过了以往任何时期, 引起了网络安全研究人员的广泛关注。

当前对 Internet 蠕虫的防范同计算机病毒一样, 主要还是依靠单机防护和局域网防火墙。由于 Internet 蠕虫的最大特点是可以通过网络自主传播, 并且在传播过程中无需人的干预, 这使得传统的防病毒措施在防范蠕虫时显得无能为力。目前国内外研究人员对 Internet 蠕虫的一致观点是应面向网络进行预警和防范, 并提出了一系列新措施。

2　蠕虫特征行为分析

只有明白了 Internet 蠕虫传播时有什么样的特征行为, 才能采取针对性措施进行有效防范。Internet 蠕虫的传播采用自动入侵技术, 受程序大小的限制, 同传统的黑客入侵相比其智能化程度不高, 传播模式也比较单一。目前蠕虫常利用的传播模式为扫描—攻击—复制。蠕虫在传播时都是针对某个特定的系统漏洞, 所以蠕虫首先要选取一台主机作为感染目标, 然后通过扫描判断该主机上是否存在能被自己利用的特定漏洞。如果漏洞存在, 蠕虫就对目标主机发动攻击, 获取一定权限, 并最终将自身复制到目标主机上完成一个传播流程。

由于蠕虫是利用某个特定的漏洞进行传播的, 蠕虫发出的扫描包内容一般都是相同的, 另外, 蠕虫在进行复制时传输的文件也是相同的。所以当有蠕虫活跃时, 网络上一般会充斥着大量内容相同的数据包。

Nicholas Weaver 曾提到蠕虫在传播时可采用预定义列表、拓扑扫描、本地子网扫描、置换扫描等方法从而大大提高感染速度[1], 但在现实中绝大多数蠕虫还是利用随机生成的 IP 地址作为感染目标。这种感染目标的不确定性决定了蠕虫在传播过程中会产生大量的针对某个端口, 但目标地址或目的端口不可达的扫描包。从蠕虫爆发过程来看, Internet 蠕虫的传播模型可以用以下流行病学传播模型来描述:

$$\frac{\mathrm{d}I_t}{\mathrm{d}t} = \beta I_t (N - I_t) \tag{1}$$

其中, I_t 表示 t 时刻感染主机数量, N 表示感染空间主机总数, β 表示相对感染率[2]。从该模型可以看出, 蠕虫的传播速度先由慢到快, 至 $I_t = N/2$ 时传播速度达到最大值, 再由快变慢直至整个感染空间趋于饱和。

根据蠕虫的特征行为和传播模型, 我们总结出蠕虫在传播时网络上具有如下特征: ①网络上充斥着大量内容相同的数据包, 甚至会严重影响网络的正常流量。这是由蠕虫传播行为的单一性决定的。②网络上被感染主机数量逐步增加, 增加过程遵循一定规律。这点可以从蠕虫的传播模型看出。③网络上会存在大量目标地址不可达或连接请求被复位的数据包, 这是由蠕虫扫描时的随机性决定的。

3　蠕虫防范技术介绍

对 Internet 蠕虫的防范可以从两方面来分析, 一是如何在第一时间发现新蠕虫; 二是怎样在第一时间采取围堵措施。目

收稿日期: 2005-08-24; 修返日期: 2005-10-29
基金项目: 国家教育部博士点基金资助项目(20020699026)

前已有的方案都是基于前面介绍的蠕虫特征行为进行防范的。下面将根据不同的原理,结合几个有代表性的实例,本文介绍国内外在蠕虫防范技术上的新思路。

3.1 内容过滤方案

内容过滤技术借鉴了对计算机病毒的防范思路,首先生成可疑数据包的签名,然后根据签名对网络流量进行过滤。但与病毒相比,蠕虫扩散的速度极快,如 Slammer 蠕虫在爆发十多分钟内感染了 Internet 上 90% 有漏洞的主机,如果还像防范计算机病毒那样依靠研究人员人工捕获蠕虫样本、分析签名、添加签名,不可能作出及时反应,因此必须研究如何无需人工干预,自动生成可疑数据包的签名的新方法。

美国 California 大学的研究人员提出了一个名为 "Early-Bird" 的网络蠕虫实时监测系统[3],其原理是在网络的边界路由器上进行流量监测,并计算网络上所有长度为 β的字符串的 Rabin 签名。如果某些数据包频繁出现在所监控的网络上,就会得到大量重复的签名。当重复次数超过某个阈值时,系统就认为网络上出现了新蠕虫,并根据得到的签名进行数据过滤。注意这里不是对蠕虫样本进行签名,而是对观测到的数据,计算其所有长度为 β的子字符串的 Rabin 签名,然后根据签名内容进行监测。在文献[3]中作者设定 β=39 取得了良好的仿真效果。

假设观测到的某段数据为 $t_1 t_2 \cdots t_n$,对于子字符串 $t_1 t_2 \cdots t_\beta$ ($\beta < n$) 的 Rabin 签名为

$$F_1 = (t_1 p^{\beta-1} + t_2 p^{\beta-2} + \cdots + t_\beta) \bmod M \quad (2)$$

其中 p 和 M 是预定义常量。对于下一个子字符串 $t_2 t_3 \cdots t_{\beta+1}$ 的 Rabin 签名可以按照如下递推算法得到:

$$F_2 = (pF_1 + t_{\beta-1} - t_1 p^{\beta-1}) \bmod M \quad (3)$$

为提高计算效率,可预先计算好 $t_i p^{\beta-1}$ 的所有可能值,以供生成签名使用。从以上分析我们可看出该系统具有以下特点:

(1) 无需获取蠕虫样本即可生成可疑流量的签名。

(2) 不是对整个蠕虫代码进行签名,而是计算给定长度的字符串的签名,这样即使蠕虫修改了部分内容逃避过滤,只要其主体内容不变仍可被检测出。

(3) 可以采用递推算法提高计算效率。

3.2 网络协议信息过滤方案

同 EarlyBird 一样,基于网络协议信息的过滤也是利用了蠕虫扫描和攻击方式的单一性,所不同的是这种防范技术不检查网络数据包的数据部分,只检查数据包的 TCP 层和 IP 层协议信息。它的工作原理是蠕虫在传播时一般都是针对某种特定的系统漏洞,因此其扫描和攻击行为也是针对目标主机的特定端口。如果在一段时间内观测到的针对某个端口的流量出现异常,即可推测当前出现了新蠕虫的传播。由于网络协议信息可以在路由器上高效获取,而且同内容过滤相比无须进行签名计算,可以大大减少计算开支,因而这种方法更具可行性。

美国 Southern California 大学的研究人员提出了一种基于路由器的蠕虫检测和围堵方案,称为 DEWP(Detector for Early Worm Propagation)[4]。其原理如图 1 所示。

DEWP 系统由蠕虫监测模块和包过滤模块两部分组成。它布置在网络的边界路由器上,可以视为在普通路由器上增添了新的功能模块。DEWP 的工作流程如下:

(1) 在一个采样周期内对进出网络的数据包进行监测,根据目的端口的不同对数据包进行分类,然后维持一张端口—连接次数关系表。如果某个曾经出现过的端口长时间没有新的连接就将其从表中清除。

(2) 对每个关系表中出现的端口,在一个采样周期内观测对该端口的访问总共涉及到多少个不同的目的地址,记为 N。计算从开始统计到现在的 N 的指数加权移动平均数 $\bar{N} = \alpha \bar{N} + (1 - \alpha) N$,这里 α是常数。

(3) 如果当前时刻的观测值 $N > (1 + \delta)\bar{N}$,则发出报警信号。这里 δ是常数,它反映了系统对流量变化的敏感度。另外此处有一个假设,就是相同时间内蠕虫对外的连接请求要大大多于正常用户,这个假设一般是成立的。

(4) 包过滤模块接到报警信号后,在网络流量中过滤掉目的端口为可疑端口的数据包。对端口的过滤有可能会影响 Internet 用户的正常访问,因而只能是暂时的,等到得到蠕虫的签名信息后可以针对内容而不是端口进行过滤。

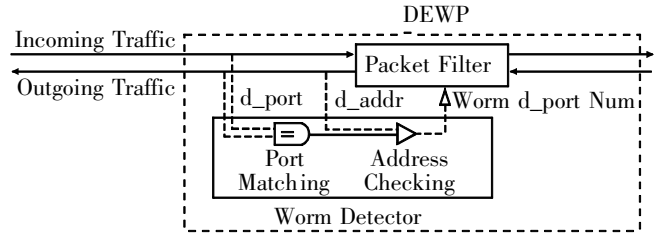

图 1 DEWP 结构框图

3.3 趋势检测方案

基于内容和协议信息过滤方案都存在一个问题,就是如何选取作为异常判断标准的阈值。如果阈值取得过低,会造成较高的虚警率,反之则会导致漏报率上升。为此研究人员又提出了一种无阈值蠕虫预警方案,其原理是检测异常情况的发展趋势,而不是异常情况出现的次数(Detecting the Trend, Not the Burst[5])。该方案的理论依据是:蠕虫传播时遵循流行病学模型,如果网络上确实有新蠕虫在传播,那么观测到的异常主机数量的增长过程将服从一定规律。通过估计 Internet 蠕虫的传播参数,可以判定当前网络上的异常主机数量增长过程是否服从流行病学模型,进而判定是否有新蠕虫的传播。

美国 Massachusetts 大学的邹长春等人提出了一种基于 Kalman 滤波的 Internet 蠕虫监测系统[5]。其结构如图 2 所示。图中在各个局域网与 Internet 相连的部分设置扫描监测器,对网络上的异常扫描信息进行统计,然后汇总到蠕虫预警中心,预警中心对各个观测器上报的统计信息进行分类和关联。通过对 Internet 蠕虫传播模型(式(1))进行分析,可以得到在蠕虫传播时,前后两个时刻的扫描包数量 Z_{t-1} 和 Z_t 满足如下关系[5]:

$$Z_t = (1 + \Delta\alpha) Z_{t-1} + v_t \quad (4)$$

其中,Δ为采样间隔,α为感染率,v_t 为与人为扫描有关的量测噪声。将式(4)作为量测方程,$X_k = [1 + \Delta\alpha]$ 作为系统参数,v_t 作为量测噪声,可以构造 Kalman 滤波器对蠕虫的感染率 α进行估计。如果当前观测到的异常扫描确实来自新蠕虫的传播,那么估计值最终会收敛于某个固定数值,否则将不会收敛。

这种针对蠕虫传播模型进行参数估计的预警无须设置阈值,通过滤波器可以有效过滤掉人为扫描对估计值的干扰,因而可以最大程度降低预警系统的漏报率和虚警率,同时又能得

到蠕虫的传播率估计值,为下一步分析蠕虫的威胁打下基础。

3.4　蠕虫传播限速方案

该方案与前面几个方案的最大不同是它并不判断网络流量是否正常,而是观测每台主机的连接失败率是否正常。如果某台主机的连接失败率超过某个阈值,则在路由器上采取一定规则丢弃该主机对外的连接请求,使其连接失败率降低到设定阈值以下。这样做的理论依据是:正常主机的对外访问请求具有确定性,而且每个用户经常访问的外部主机是有限的,因而很少出现连接失败;蠕虫在确定攻击目标时具有随机性,所选取的对象经常不存在或者没有启动相应的服务,因而很容易出现连接失败现象。如果某台主机的连接失败频率高于某个阈值,则假定该主机感染了蠕虫,通过在路由器上限制该主机对外连接的频率可以降低该主机对外扫描的频率,这就相当于降低了蠕虫的感染率,从而为人们采取对抗措施赢得了时间。

美国 Florida 大学的 Chen Shigang 等人提出了一种应用于 ISP 的蠕虫防范系统 DAW(Distributed Anti Worm) ,它由部署在 ISP 的边界路由器上的 DAW 代理(DAW Agent) 和管理中心(Management Station) 组成。其结构如图 3 所示[6]。

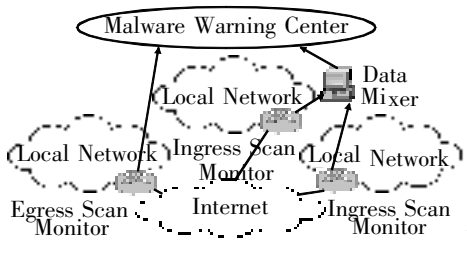

图 2　Internet 蠕虫预警系统结构图

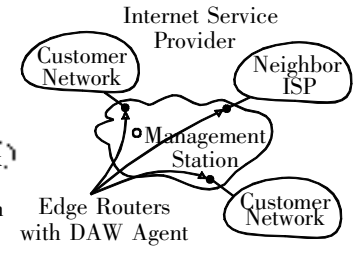

图 3　DAW 结构图

图中每个 DAW 代理维护一张 Hash 表,Hash 表的关键字为所属网络内主机的 IP 地址。每当 DAW 代理收到一个从其他网络发送到本网络的连接失败应答时,它就根据应答包的地址更新 Hash 表内相应记录的连接失败率。如果某个主机的连接失败率超过预先设定的阈值,则采用令牌桶算法在路由器上丢弃该主机的部分连接请求。DAW 代理将自己的观测数据发往管理中心,如果管理中心汇总的报告显示目前连接失败异常的主机稳步增加,则有理由相信当前网络中出现了蠕虫的传播。

采用蠕虫传播限速方案可以减缓蠕虫的传播速度,为人们采取围堵措施赢得时间。它可以发现当前已染主机,限定这些主机的连接活动,同时不影响正常主机的访问。

3.5　良性蠕虫对抗方案

良性蠕虫对抗方案是一种新的蠕虫防范思路,它借鉴了蠕虫传播的思想来对抗恶性蠕虫。当网络上爆发了针对某种漏洞的蠕虫时,我们构造并传播对应的良性蠕虫。良性蠕虫也像其他蠕虫一样在网上扩散,发现未打补丁的主机就“感染”该主机。如果该主机未被恶性蠕虫感染,就为该主机打上补丁,使其免受恶性蠕虫入侵;如果该主机已被感染,就先清除恶性蠕虫,然后为其打上补丁。

目前蠕虫对抗技术有很多困难,一方面,很多蠕虫感染目标主机后就为其打上了补丁,这样良性蠕虫就很难扩散到该主机;另一方面,良性蠕虫的扫描速度、扩散速度和范围都要经过精心设计,而且一定要有一套自我控制机制,否则一旦传播到网上可能会同恶性蠕虫一样引起网络瘫痪。2001 年出现的 Cheese 蠕虫就是为了查杀 Lion 蠕虫而出现的,它利用了 Lion 蠕虫留下的后门进行传播,虽然其本意是好的,但由于其传播时要发送大量扫描包同样会造成网络堵塞。

除了前面提到的几种 Internet 蠕虫防范技术,还有研究人员提出了其他一些蠕虫检测方法。例如 Xinzhou Qin 等人提出了一种利用局域网进行蠕虫发现的方案[8], Brent N. Chun 等人提出了一种分布式蠕虫检测方案 Netbait[9]。这些方案也都是利用分布于网络上的检测点搜集异常信息,然后送至检测中心进行分析和判断,与前面介绍的几种方案有相似之处,限于篇幅本文不再详述。

4　总结

由于 Internet 缺乏中心控制能力和蠕虫的自主传播特性,导致目前蠕虫频频爆发,而 Internet 用户缺少有效的防治对策。当前依靠单机防护软件和局域网防火墙进行过滤的防范措施具有明显的滞后性,不能从根本上扭转在 Internet 蠕虫攻防中的被动局面。网络的问题还应该依靠网络来解决,应构建面向网络的 Internet 蠕虫防范系统。

本文首先分析了蠕虫传播的特征行为和传播模型,然后重点介绍了国内外在 Internet 蠕虫防范技术上的最新进展。需要指出的是,计算机蠕虫本质上是入侵行为的自动化,更多的黑客技术将会被用到蠕虫编写中来,蠕虫技术也在不断地完善和发展。随着蠕虫编写技术的成熟,从漏洞发现到出现利用该漏洞的蠕虫之间的时间差越来越短,这对我们的应变能力提出了更高的要求,对蠕虫的防治和对抗将是长期而困难的工作。

参考文献:

[1]　icholas Weaver. Potential Strategies for High Speed Active Worms: A Worst Case Analysis [EB/OL]. http://www.cs.berkeley.edu/～nweaver/worms.pdf, 2002.

[2]　David Moore, Colleen Shannon, Jeffery Brown. Code-Red: A Case Study on the Spread and Victims of an Internet Worm [EB/OL]. http://www.caida.org/outreach/papers/2002/codered.pdf, 2002.

[3]　S Singh, C Estan, G Varghese, *et al*. The Earlybird System for Real-time Detection of Unknown Worms [R]. Technical Report CS2003-0761, University of California, San Diego, 2003.

[4]　Chen Xuan, Heidemann John. Detecting Early Worm Propagation through Packet Matching [R]. Technical Report ISI-TR-2004-585, USC/Information Sciences Institute, 2004.

[5]　Zou Cliff C, Gong Weibo, Don Towsley, *et al*. Monitoring and Early Detection for Internet Worms [EB/OL]. http://tennis.ecs.umass.edu/～czou/research/earlyDetectionJournal.pdf, 2003.

[6]　Chen Shigang, Tang Yong. Slowing Down Internet Worms [EB/OL]. http://www.cise.ufl.edu/～sgchen/papers/icdcs 2004.pdf, 2004.

[7]　云晓春. Internet 蠕虫主动遏制[C]. 北京:全国网络与信息安全技术研讨会专题报告, 2004.

[8]　Qin Xinzhou, David Dagon, Gu Guofei, *et al*. Worm Detection Using Local Networks [EB/OL]. http://www.cc.gatech.edu/people/home/xinzhou/TR_CoC_04.pdf, 2003.

[9]　Brent N Chun, Jeason Lee, Hakim Weatherspoon. Netbait: A Distributed Worm Detection Service [R]. Intel Technical Report IRB-TR-03-033, 2003.

(下转第 30 页)

进而降低规则的置信度来隐藏规则。

对实验结果的衡量包括三个参数: 损失规则的比率(*LR*) , 增加规则的比率(*AR*) , 噪音率(*NR* = *LR* + *AR*) 。由于数据集的相似度较高, Support/Confidence 设置为 0.6/0.9, 共产生了 57 492条规则, 从中选取五条规则进行隐藏。

实验结果如表 1 所示。

表 1 三种算法的对比

参数	SWA	Algo2a	OSA
LR	1.306%	1.929%	1.689%
AR	1.925%	0.02%	0.115%
NR	3.232%	1.949%	1.804%

从表 1 可以得出以下结论: 在规则损失方面, OSA 介于 SWA 算法和 Algo2a 之间; 新增规则方面, OSA 和 Algo2a 相近, 大大优于 SWA 算法; 噪音率则是 OSA 算法最优。综合分析, OSA 算法要优于 SWA 和 Algo2a, 是对上述两种算法的综合和改进。

6 结束语

本文对关联规则的隐藏保护技术进行了一定的探讨, 对目前的关联规则隐藏算法进行了介绍和分析, 并提出了一个改进的关联规则隐藏算法 OSA。实验结果表明该算法是一种优良的关联规则隐藏算法, 其综合性能要高于大多数现行的算法。

参考文献:

[1] E O'Leary. Knowledge Discovery as a Threat to Database Security [C]. Proc. of the 1st Int'l Conf. Knowledge Discovery and Databases, 1991. 107-516.

[2] C Clifton, D Marks. Security and Privacy Implications of Data Mining [C]. Proc. of ACM Workshop Data Mining and Knowledge Discovery, 1996.

[3] M Atallah, E Bertino, A Elmagarmid, *et al*. Disclosure Limitation of Sensitive Rules[C]. Chicago: Proc. of IEEE Knowledge and Data Engineering Workshop, 1999. 45-52.

[4] A Evfimievski, R Srikant, R Agrawal, *et al*. Privacy Preserving Mining of Association Rules[C]. Edmonton: Proc. of the 8th ACM SIGKDD Int. Conf. on Knowlegde Discovery and Data Mining, 2002. 217-228.

[5] S R M Oliveira, O R Za ane. Privacy Preserving Frequent Itemset Mining[C]. Maebashi City: Proc. of the IEEE ICDM Workshop on Privacy, Security, and Data Mining, 2002. 43-54.

[6] S R M Oliveira, O R Za ane. Protecting Sensitive Knowledge by Data Sanitization[C]. Melbourne: Proc. of the 3rd IEEE International Conference on Data Mining (ICDM'03), 2003. 613-616.

[7] Oliveira S R M, Zaane O R, Saygin Y. Secure Association Rule Sharing[A]. Dai H, Srikant R, Zhang Cs. Advances in Knowledge Discovery and Data Mining[C]. Sydney: The 8th Pacific-Asia Conference, Proceedings, volume 3056 of Lecture Notes in Artificial Intelligence, 2004. 74-85.

[8] Vassilios S Verykios, Ahmed K Elmagarmid, Elisa Bertino, *et al*. Association Rule Hiding[J]. IEEE Transactions on Knowledge and Data Engineering, 2004, 16(4).

[9] E Dasseni, V S Verykios, A K Elmagarmid, *et al*. Hiding Association Rules by Using Confidence and Support[C]. Pittsburgh: Proc. of the 4th Information Hiding Workshop, 2001. 369-383.

[10] Y Saygin, V S Verykios, C Clifton. Using Unknowns to Prevent Discovery of Association Rules[J]. SIGMOD Record, 2001, 30(4): 45-54.

[11] R Agrawal, T Imielinski, A Swami. Mining Association Rules Between Sets of Items in Large Databases[C]. Washington, DC: Proc. of ACM-SIGMOD Int. Conf. Management Data(SIGMD'93), 1993.

作者简介:

丁小刚(1979-), 男, 硕士, 主要研究领域为数据库安全、数据挖掘、数据仓库; 黄伟伟, 男, 硕士, 研究方向为数据挖掘、数据库安全; 柏文阳(1967-), 男, 副教授, 研究方向为数据库安全、数据挖掘、数据仓库。

(上接第 15 页)

[10] pafford Eugene H. The Internet Worm Program: An Analysis [J]. ACM Computer Communication Review, 1989, 19(1): 17-57.

[11] 郑辉, 李冠一, 涂奉生. 蠕虫的行为特征描述和工作原理分析[C]. 第三届中国信息和通信安全学术会议论文集. 北京: 科学出版社. 2003. 168-172.

[12] 栾新民, 廖闻剑. Nimda 蠕虫分析与防范[J]. 计算机应用研究, 2002, 19(11): 155-158.

[13] Zou Cliff C, Gong Weibo, Don Towsley. Code Red Worm Propagation Modeling and Analysis [C]. Washington, DC: CCS'02, 2002. 18-22.

[14] Lance Spitzner. Honeypots, Definitions and Value of Honeypots [EB/OL]. http://www.spitzner.net/honeypots.html, 2001.

[15] 卿斯汉, 文伟平, 蒋建春, 等. 一种基于网状关联分析的网络蠕虫预警新方法[J]. 通信学报, 2004, 25(7): 62-70.

[16] 邱晓鹏, 张玉清, 冯登国. 蠕虫攻防技术综述[C]. 全国网络与信息安全技术研讨会专题论文集, 2004. 57-62.

[17] Marc Mazuhelli. A Virus and a Worm: Lessons Learned from Sircam and Code Red in a University Environment [EB/OL]. http://rr.sans.org/malicious/sircam2.php, 2003.

[18] Jose Nazario, Jeremy Anderson, Rick Wash, *et al*. The Future of Internet Worms[EB/OL]. http:// www.blackhat.com/presentations/JoseNazario/bh-usa-01-Joes-Nazario.pdf, 2002.

[19] Jason C Hung, Kuan-Cheng Lin, Anthony Y Chang. A Behavior-based Anti-Worm System[EB/OL]. http://doi.ieeecomputersociety.org/10.1109/AINA.2003.1193006, 2003.

[20] 左晓栋, 戴英侠. "狮子"蠕虫分析及相关讨论[J]. 计算机工程, 2002, 28(1): 16-17.

[21] Zesheng Chen, Lixin Gao, Kevin Kwiat. Modeling the Spread of Active Worms[EB/OL]. http://www.ieee-infocom.org/2003/papers/46_03.pdf, 2003.

[22] Stuart E Schechter, Jaeyeon Jung, Arthur W Berger. Fast Detection of Scanning Worm Infections[EB/OL]. http://eecs.harvard.edu/~stuart/papers/scanworm.pdf, 2003.

[23] Stelios Sidiroglou, Angelos D Keromytis. Countering NetworkWorms through Automatic Patch Generation[C]. IEEE Security and Privacy, 2005.

[24] Manuel Costa, Jon Crowcroft, Miguel Castro, *et al*. Can We Contain Internet Worms[EB/OL]. http://research.microsoft.com/~antr/MS/HotNetsVigilante.pdf, 2003.

[25] Shigang Chen, Sanjay Ranka. An Internet Worm Early Warning System[EB/OL]. http://www.cise.ufl.edu/~sgchen/papers/globecom2004_worm.pdf, 2004.

作者简介:

周涛(1979-), 男, 博士研究生, 主要研究方向为网络环境下复杂系统控制与信息安全; 戴冠中(1937-), 男, 教授, 博士生导师, 主要研究方向为自动控制、网络信息安全; 慕德俊(1963-), 男, 教授, 博士生导师, 主要研究方向为模式识别、网络信息安全。