基于隔离策略的蠕虫传播模型及分析*)

张运凯[1,2] 王方伟[2] 马建峰[1] 张玉清[3]

(西安电子科技大学计算机学院 西安710071)[1] (河北师范大学网络信息中心 石家庄050016)[2]

(中科院研究生院国家计算机网络入侵防范中心 北京100039)[3]

摘 要 近几年,蠕虫频繁爆发,而且传播愈来愈快,破坏力也增大,已成为互联网安全的主要威胁。基于经典的Kermack-Mckendrick模型,本文提出了一个采用动态隔离策略、动态传染率和恢复率的蠕虫传播模型。分析表明此模型能更有效降低蠕虫的传播速度,为防御蠕虫赢得更多宝贵的时间,减缓和降低了蠕虫造成的负面影响,仿真试验证明了此模型的有效性。

关键词 蠕虫传播,隔离,SARS,传染病模型,网络安全

Worm Propagation Modeling and Analysis Based on Quarantine

ZHANG Yun-Kai[1,2] WANG Fang-Wei[2] MA Jian-Feng[1] ZHANG Yu-Qing[3]

(Computer Science and Technology, Xidian University, Xi'an710071)[1] (Network Center, Hebei Normal University, Shijiazhuang050016)[2]

(National Computer Network Intrusion Protection Center, GSCAS, Beijing100039)[3]

Abstract In recent years, the worms that had a dramatic increase in the frequency and virulence of such outbreaks have become one of the major threats to the security of the Internet. In this paper, we provide a worm propagating model. It bases on the classical epidemic Kermack-Kermack model, adopts dynamic quarantine strategy, dynamic infecting rate and removing rate. The analysis shows that model can efficiently reduce a worm's propagation speed, which can give us more precious time to defend it, and reduce the negative influence of worms. The simulation results verify the effectiveness of the model.

Keywords Worm propagation, Quarantine, SARS, Epidemic model, Network security

1 引言

现在计算机蠕虫越来越容易传播,已经对互联网的可靠性构成了很大的威胁。特别是在高速网络环境下,多样化的传播途径和应用环境更使得蠕虫爆发率大为提升,覆盖面也更加广泛,造成的损失越来越大。蠕虫攻击可以由任何人发动,可以来自世界任何地方,蠕虫传播时间愈来愈短,隐蔽性也越强,例如:"Slammer"蠕虫大小只是"Code Red"的十分之一,但前者只需10分钟就能传遍全球,而后者需要三天时间,蠕虫的传播时间由原来的数天降到了数分钟。Probe Group of Cedar Knolls组织的一位分析家认为:未来蠕虫的传播时间仅以秒计。如flash蠕虫能在30秒内感染所有存在漏洞的主机,能在几百毫秒内感染一个企业的所有主机[1]。从第一个蠕虫——Morris蠕虫爆发到现在已有16年的时间,其间出现过很多种类的蠕虫,这从侧面说明目前的网络很脆弱,而且Weaver[2]还提出了一些能使蠕虫传播更快的设计原则,这更加重了防治的困难。为了能更好地防御蠕虫,就必须理解蠕虫的传播行为,理解蠕虫的不同属性(传播模式、对网络流量的影响、网络拓扑结构等),准确模拟蠕虫的传播过程。

一个精确的蠕虫传播模型可以使人们对蠕虫有更清楚的认识,能确定其在传播过程中的弱点,而且能更精确地预测蠕虫所造成的损失。目前已有很多学者对蠕虫进行了深入研究,提出了一些模型。传统的蠕虫传播模型都是均匀的,即每台主机等概率地感染其它易感主机[3~6]。考虑到蠕虫本地交互作用,Kephart[7]把传染病模型扩展到非均匀网络:随机图、2维晶格和类树层次图,因为主机间主要考虑磁盘共享,所以不能很有效地模拟蠕虫传播。而且Kephart等人采用SIS(易感—感染—易感)模型,假定每一台主机可以被重复感染,它不适合模拟单一蠕虫的传播,因为通过打补丁或清除,主机会对此蠕虫有免疫作用。Staniford等[1]采用经典、简单模型对"Code Red"进行了仿真,仿真结果和实际观察的数据匹配得很好。Moore[8]提供了一些有价值的数据,并且对蠕虫的行为进行了深入的分析。Cliff Changchun Zou[3]提出了二因素传播模型,并利用该模型分析了Code Red蠕虫的传播,研究了人为防范对蠕虫传播的影响。研究人员从网络扫描的各种方法入手,研究蠕虫的传播机制。Cliff Changchun Zou还研究了网络蠕虫的预警模型,提出了采用Kalman过滤法在蠕虫传播初期实时检测的方法。David Moore[4]研究了响应时间、响应策略和配置策略对蠕虫传播的影响。Cliff Changchun Zou[9]针对企业网提出了一种基于防火墙网络系统的蠕虫防治方法,但不适合于采用对等网(peer-peer)的企业,存在一定的局限性。Cliff Changchun Zou[10]提出了一个基于动态隔离措施的模型,采用"判定无罪之前先假定有罪"的思想。很多模型的蠕虫扫描率都假定为一常数,这与实际情况不符,因为大规模的蠕虫传播会导致网络拥塞,致使网络路由器的性能下降,从而会降低蠕虫扫描率,所以应该采用动态扫描率。另外,考虑蠕虫的清除时,不能只计算感染主机的清除,还应计算易感主机,因为随着蠕虫的传播,人们对蠕虫的认识会更清楚,防御

*)本文得到国家信息关防与网络安全保障持续发展计划、国家863高技术研究发展计划(2002AA142151)、中国科学院知识创新工程方向性项目(KGCX2-106)、北京市科技计划项目(H020120090530)资助。张运凯 博士研究生,副教授,主要研究方向为网络安全。王方伟 硕士,主要研究方向为网络安全。马建峰 教授,博士生导师,主要研究方向为信息安全。张玉清 副研究员,博士后,主要研究方向为网络与信息系统安全。

意识会增强，人们会自然而然地考虑到对所有易感主机采取保护措施，如打补丁、升级、建立过滤器以阻止蠕虫的扫描、断开某些重要易感主机的网络连接。还有一个很重要的因素是恢复率，目前所见的文献中都假定为一常数，这也与实际不符，就像现实生活中的传染病(如SARS)一样，开始人们对它认识不足，不知道它的传播途径及发病机理，只能摸索着采取一些措施，随着人们认识的提高，会采取相应的措施，能更有效地治疗此类病，所以蠕虫的恢复率也应是变量。针对已有模型存在的问题，本文提出了一个新的模型，它具有以下特点：(1)采用防治SARS传染病一样的动态隔离措施，通过禁止某些主机出现异常流量的端口，来隔离主机；(2)采用动态感染率；(3)采用动态恢复率。

2 传统的蠕虫传播模型

简单计算机蠕虫和生物病毒在自复制和传播等方面有很多相似之处，所以也可以用数学方法来研究计算机蠕虫。在流行病学领域，有两种模型来模拟传染疾病的传播：随机模型和确定模型。前者适合有简单病毒的小规模系统，后者适合有复杂病毒的大规模系统。因为蠕虫的传播是全球性的，所以采用确定模型。目前存在很多数学模型，本文首先介绍其中的两种经典模型[3,10]，虽然存在一些缺陷，如没有考虑用户所采取的防治措施和感染率 β 是常数等，但这两个是其他模型[5,6,9,10]的基础，很多模型都是由这两种得到的，如文[9,10]。为了便于描述，先解释一下本文中用到的符号，详见表1。

表1 符号参数说明

符号	定义
$I(t)$	在时刻 t 被感染的主机数
$S(t)$	在时刻 t 被怀疑为疑似的主机数
$R_1(t),G_1(t)$	在时刻 t 从易感主机中恢复、隔离的主机数
$R_2(t),G_2(t)$	在时刻 t 从感染主机中恢复、隔离的主机数
N	主机总数
$\gamma_0,\gamma_1(t),\gamma_2(t)$	蠕虫传播模型中的恢复率
Ω	蠕虫扫描空间
$\beta_0,\beta'(t),\beta''$	蠕虫传播模型中的感染率
ρ,ρ',ρ''	蠕虫爆发阈值
p_1,p_2	易感、感染主机的隔离概率
η	蠕虫扫描率
T	动态隔离时间

2.1 经典的、简单的流行病模型

在此模型中，每一台主机只有两种状态：疑似或感染，而且一旦被传染就永远保持感染状态，其状态转移只能表示为：易感→感染，其模型表示为：

$$\frac{dI(t)}{dt}=\beta_0 I(t)[N-I(t)] \tag{1}$$

当 $t=0$ 时，有 $I(0)$ 个已感染的主机，有 $N-I(0)$ 个疑似的主机。

2.2 Kermack-Mckendrick 模型

在经典的、一般的流行病：Kermack-Mckendrick 模型中考虑了感染主机的恢复，它假定在发病期间，一些受感染的主机可以恢复为正常状态或死亡，且对此传染病具有免疫功能，因此每个主机具有三种状态：疑似、感染或恢复，其状态转移可表示为易感→感染→恢复，或永远保持易感状态。基于上面的简单模型(1)得到 Kermack-Mckendrick 模型，表示为：

$$\begin{cases}\dfrac{dI(t)}{dt}=\beta_0 I(t)S(t)-\gamma_0 I(t)\\ \dfrac{dR_1(t)}{dt}=\gamma_0 I(t)\\ N=I(t)+R_1(t)+S(t)\end{cases} \tag{2}$$

从 Kermack-Mckendrick 模型得出了一个重要的理论—传染病爆发定理：一个大规模蠕虫爆发的充分必要条件是初始疑似主机的数目 $S(0)>\rho$。

3 基于隔离策略的蠕虫传播模型

互联网上真实蠕虫的传播是一个很复杂的过程，为了方便起见而又不失一般性，本文只考虑连续活动的蠕虫，而且没有考虑网络拓扑限制，网络拓扑限制是指感染主机不能直接感染任意易感主机，它需要先感染一些中间主机才能到达目标主机。很多蠕虫(如"Code Red")都与网络拓扑无关，但邮件病毒(Melissa 和爱虫)是依赖于网络拓扑结构的，关于这部分本文不作深入讨论。

蠕虫传播速度非常快，单靠人工防御收效甚微，必须有一个自动防御系统。一旦有主机被感染，自动防御系统立即将该主机隔离，再由安全组的成员去"治疗"，以防蠕虫进一步传播。对于未知的蠕虫则只能借助于异常检测方法，如果某主机出现异常，立即隔离该主机。异常检测不可避免地出现误警，为了不影响用户的正常使用，隔离的时间不能过长。

3.1 模型一：基于隔离策略的 Kermack-Mckendrick 蠕虫传播模型

在 Kermack-Mckendrick 模型中，每台主机有三种状态：易感、感染和恢复，从实际情况考虑，主机的恢复应包括两部分：感染主机的恢复和易感主机的恢复。在某一时刻，主机处于隔离或非隔离状态，只居其一，这三者之间的关系如图1所示。

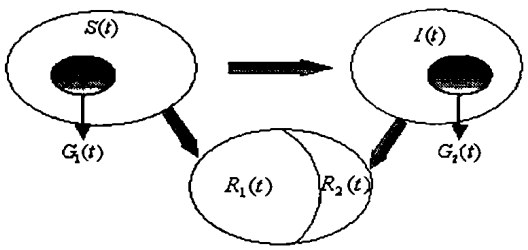

图1 易感、感染、恢复主机三者间的关系

定义 T 为隔离时间，$R_1(t)$、$G_1(t)$分别表示到时刻 t 从易感主机中恢复、隔离的主机数，$R_2(t)$、$G_2(t)$表示到时刻 t 从感染主机中恢复、隔离的主机数。定义 λ_1 为易感主机的隔离率，即一台正常的易感主机在被隔离之前还能平均活动 $1/\lambda_1$ 个单位时间，λ_1 越大，异常检测系统的误报率越大，系统的性能越低。λ_2 为感染主机的隔离率，即一台感染主机在被隔离之前能平均活动 $1/\lambda_2$ 个单位时间，λ_2 越大，异常检测系统的检测率越好。定义 γ_1 为易感主机的恢复率，γ_2 为感染主机的恢复率，随着人们对蠕虫的认识提高，治疗更有针对性，所以这两个恢复率也都是变量，分别表示为 $\gamma_1(t)$ 和 $\gamma_2(t)$。通过采取保护措施，如打补丁、升级、建立过滤器以阻止蠕虫的扫描、断开某些重要易感主机的网络连接等措施，蠕虫的有效感染率 β_0 会降低，也应为变量，表示为 $\beta(t)$。假设在恢复过程中，采用均匀恢复，即所有的主机等概率恢复。在 t 时刻，$G_1(t)$中所有已隔离了 T 时间的主机被释放。对于任意时间 τ，只有 $S(\tau)-G_1$

(τ)个易感主机没有被隔离，由于只考虑等概率恢复，因此从$G_1(\tau)$中恢复的主机数为$\gamma_1 G_1(\tau)$，所以可以得到下式：

$$G_1(t)=\int_{-T}[S(\tau)-G_1(\tau)]\lambda_1 d\tau-\int_{-T}\gamma_1(t)G_1(\tau)d\tau \tag{3}$$

由于隔离时间T很小，在$t-T$到t间，$S(t)$和$G_1(t)$变化不大，根据大数法则，可以近似地认为不变，由此可以得到下式：

$$\begin{cases}S(\tau)\cong S(t)\\ G_1(\tau)\cong G_1(t)\end{cases}\quad \forall\ \tau\in[t-t,t] \tag{4}$$

由式(3)、(4)可以得下式：

$$G_1(t)=p_1S(t)，其中\ p_1=\frac{\lambda_1 T}{1+(\lambda_1+\gamma_1(t))T} \tag{5}$$

同理可以得到$I(t)$和$G_2(t)$之间的关系，

$$G_2(t)=p_2I(t)，其中\ p_2=\frac{\lambda_2 T}{1+(\lambda_2+\gamma_2(t))T} \tag{6}$$

根据 Kermack-Mckendrick 模型式(2)，从时刻t到$t+\Delta t$，易感主机数增长为：

$$S(t+\Delta t)-S(t)=-\beta(t)[S(t)-G_1(t)][I(t)-G_2(t)]\Delta t-\gamma_1(t)S(t)\Delta t$$

结合(5)、(6)式所以得到：

$$\frac{dS(t)}{dt}=-\beta'(t)S(t)I(t)-\gamma_1(t)S(t) \tag{7}$$

其中$\beta'(t)=(1-p_1)(1-p_2)\beta_0(t)$，由于$p_1$和$p_2$都小于1，因此蠕虫的传染率减小了$(1-p_1)(1-p_2)$倍，新模型的蠕虫爆发阈值为：

$$\rho'=\gamma_2(t)/\beta'=\frac{1}{(1-p_1)(1-p_2)}\frac{\gamma_2(t)}{\rho} \tag{8}$$

因为$\gamma_2(t)$在增大，所以新模型的蠕虫爆发阈值至少增大了$1/(1-p_1)(1-p_2)$倍，从而减少了蠕虫爆发的可能。同理可得：

$$\frac{dI(t)}{dt}=-\beta'(t)S(t)I(t)-\gamma_2(t)I(t) \tag{9}$$

所以基于隔离策略的 Kermack-Mckendrick 蠕虫传播模型为：

$$\begin{cases}\dfrac{dS(t)}{dt}=-\beta'(t)I(t)S(t)-\gamma_1(t)S(t)\\ \dfrac{dI(t)}{dt}=\beta'(t)I(t)S(t)-\gamma_2(t)I(t)\\ \dfrac{dR_1(t)}{dt}=\gamma_1(t)S(t)，\dfrac{dR_2(t)}{dt}=\gamma_1(t)I(t)\\ N=I(t)+S(t)+R_1(t)+R_2(t)\end{cases} \tag{10}$$

为了便于和文[10]中的模型作比较，现将其(10)式的模型表示为模型 A：

$$\frac{dI(t)}{dt}=\beta''I(t)S(t)-\gamma_2 I(t)$$

其中，$\beta''=(1-\frac{\lambda_1 T}{1+\lambda_1 T})(1-\frac{\lambda_2 T}{1+(\lambda_2+\gamma)T})\beta$。(为了和本文中的符号统一，做了相应的调整。)

3.2 模型二：只考虑隔离主机清除的蠕虫传播模型

在上面的模型中，所有被隔离的主机都是等概率被恢复，但往往由于安全组成员数目有限，不可能对所有的主机都照顾到，在这种情况下，比较可行的方法是仅“治疗”感染主机中被隔离和易感主机中因发生异常流量而被隔离的那些主机，所以这两部分被恢复的主机数分别为：

$$\begin{cases}\dfrac{dR_1(t)}{dt}=\gamma_1(t)G_1(t)\\ \dfrac{dR_2(t)}{dt}=\gamma_2(t)G_2(t)\end{cases} \tag{11}$$

新的模型为：

$$\begin{cases}\dfrac{dS(t)}{dt}=-\beta'(t)I(t)S(t)-\gamma_1'(t)S(t)\\ \dfrac{dS(t)}{dt}=-\beta'(t)I(t)S(t)-\gamma_2'(t)S(t)\\ \dfrac{dR_1(t)}{dt}=\gamma_1(t)G_1(t)，\dfrac{dR_2(t)}{dt}=\gamma_2(t)G_2(t)\\ N=I(t)+S(t)+R_1(t)+R_2(t)\end{cases} \tag{12}$$

其中$\gamma_1'(t)=p_1\gamma_1(t)$，$\gamma_2'(t)=p_1\gamma_2(t)$，此模型的蠕虫爆发阈值为：

$$\rho''=\gamma_2'(t)/\beta'=\frac{p_2}{(1-p_1)(1-p_2)}\frac{\gamma_2(t)}{\rho} \tag{13}$$

如果开始易感主机数$S(0)<\rho''$，蠕虫就不会爆发，从而降低了蠕虫爆发的机会。

为了便于和文[10]中的模型作比较，现将其(12)式的模型表示为模型 B：

$$\frac{dI(t)}{dt}=\beta''I(t)S(t)-\gamma_2 R(t)$$

4 仿真试验及分析

上面这两个模型中都包括好几个不确定的动态参数，如$\beta_0(t)$，$G_1(t)$，$G_2(t)$，$\gamma_1(t)$，$\gamma_2(t)$，所以不能得到其精确的解析解，试验时采用 Matlab Simulink 工具进行仿真，为了证明本模型的有效性，采用两个试验环境。首先采用和 Cliff Changchun Zou[10]的模型完全相同的试验环境，以方便进行对比：假定主机总数$N=75000$，蠕虫平均扫描率$\eta=4000$/秒，初始感染主机数$I(0)=10$，假定某主机被警告的时间服从指数分布，易感主机的隔离率$\lambda_1=0.00002315$/秒，即一台正常的易感主机平均每天被隔离2次，感染主机的隔离率$\lambda_2=0.2$/秒，即一台感染主机在被隔离之前能传播5秒，隔离时间$T=10$秒。

文[10]中的传染率$\beta(t)=\beta_0[1-I(t)/N]^\eta$，它只考虑了目前已被感染的主机，而实际人们的“医疗”水平跟所有感染的主机数有关，不管恢复有否，所以本文的蠕虫传染率定义为$\beta(t)=\beta_0[1-J(t)/N]^\eta$，其中$J(t)$为到时刻$t$所有被感染的主机数，$J(t)=I(t)+R_2(t)$，指数$\eta$为一非负实数，用来调整传染率对$J(t)$的敏感程度，设$\eta=3$，$\beta_0=\eta/\Omega=9.3\times10^{-7}$表示初始传染率。

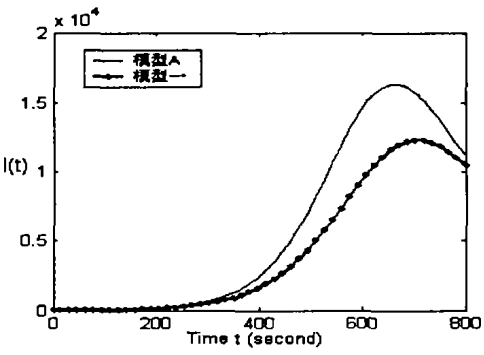

图2 基于隔离的 Kermack-Mckendrick 蠕虫传播模型的比较

蠕虫恢复率$\gamma_1(t)=\gamma_1^0[1+R_1(t)/N]^\alpha$，$\gamma_1^0=0.0001$为易感

主机的初始恢复率，恢复的易感主机数越多，$\gamma_1(t)$越大，跟实际情况相符；类似地，$\gamma_2(t)=\gamma_2^0[1+R_2(t)/N]^\phi$，其中 $\gamma_2^0=0.01$ 为感染主机的初始恢复率，ω,ϕ 都为非负实数，设 $\omega=3,\phi=3$。

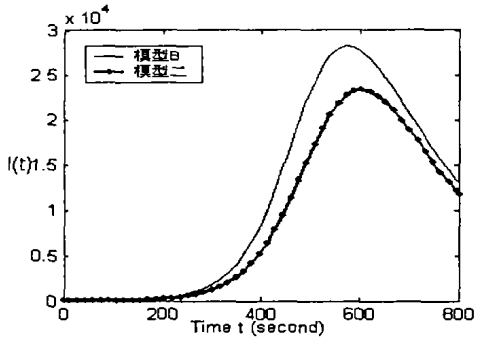

图3　只考虑隔离主机清除的蠕虫传播模型的比较

第二个试验环境：$N=36000,\eta=358$/分，$\gamma_1^0=0.00001,\gamma_2^0=0.001$，其余参数数值和第一个试验环境相同，单位为分钟。

从图2、图3可以清楚地看出，本文模型的蠕虫到达高峰期的感染主机数明显低于文[10]中相应的模型，并且推迟了到达高峰期的时间。第二个试验环境的蠕虫扫描率比较小，感性地看，图4、图5到达高峰期的时间差不多，但还是能明显地看出高峰期时感染主机数要小于文[10]中的模型，下降的趋势也比较快。下面再从定量的角度进行比较，详见表2。

从表2中可以看出，试验环境一本文模型比文[10]模型高峰期时感染主机数平均减少4447.5，时间平均推迟了40秒，试验环境二中本文模型比文[10]模型高峰期时感染主机数平均减少15975，时间平均推迟了37分钟。

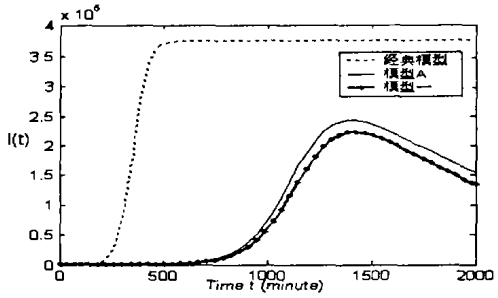

图4　基于隔离的 Kermack-Mckendrick 蠕虫传播模型的比较

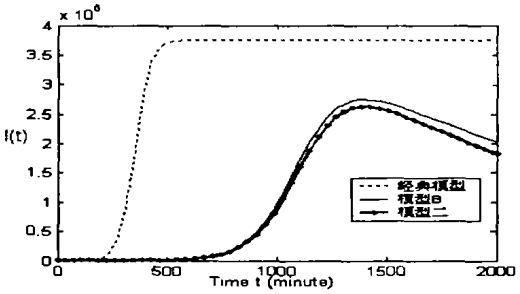

图5　只考虑隔离主机清除的蠕虫传播模型的比较

表2　本文模型和文[10]相应模型的性能比较

试验环境 \ 项目		模型A	模型一	模型B	模型二
一	到高峰期的时间(秒)	670.42	718.42	571.49	603.49
	高峰期时感染总数	16304	12324	28291	23376
二	到高峰期的时刻(分)	1386	1424	1366	1402
	高峰期时感染总数	242700	222860	274410	262300

结论　本文提出了一个新的基于隔离策略的蠕虫传播模型，其特点是：

(1)采用动态隔离策略，如果某易感主机出现异常，就把该主机隔离，以便安全组成员“诊断”，由于蠕虫检测系统存在误警，隔离时间不能过长；将感染主机进行隔离并“治疗”，以防止蠕虫的进一步传播，从而有效地降低了蠕虫的传染率。

(2)采用动态传染率、动态恢复率，随着蠕虫的传播，人们防治蠕虫的意识、手段会不断提高，自然而然地会采取一些更有针对性地措施，所以蠕虫的有效传染率会降低，恢复率会增大。

仿真试验结果表明，该模型能有效地降低蠕虫的传播速度，减少蠕虫爆发的可能。

下一步的工作主要包括：隔离时间的长短对本模型的影响；研究非均匀系统下蠕虫的传播模型；研究延时条件下蠕虫的传播模型。

参考文献

1 Staniford S, Paxson V, Weaver N. How to own the Internet in your spare time. In: Proc. of the USENIX Security Symposium, 2002. 8: 149～167

2 Weaver N. Warhol worms: the potential for very fast Internet plagues, http://www.cs.berkeley.edu/～nweaver/warhol.html

3 Zou Cliff Changchun, Gong W, Towsley D F. Code red worm propagation modeling and analysis. In: ACM Conf. on Computer and Communications Security, 2002. 138～147

4 Moore D, Shannon C, Geoffrey et al. Internet quarantine: Requirements for containing self-propagating code. INFOCOM 2003. Available: http://www.cs.ucf.edu/～jglenn/research.html

5 Zou Cliff Changchun, Towlsey Don, Gong W. On the performance of Internet worm scanning strategies: [Umass ECE Technical Report TR-03-CSE-07]. 2003, 9: 1～15

6 Wang Yang, Wang Chenxi. Modeling the effects of timing parameters on virus propagation, In: Proc. of the 2003 ACM workshop on Rapid Malcode, 2003. 10: 61～66

7 Kerhart J O, Chess D M, White S R. Computers and epidemiology, IEEE Spectrum, 1993. 20～26

8 Moore D. The spread of the code-red worm, http://www.caida.org/analysis/security/code-red/coderedv2_analysis.xml

9 Zou Cliff Changchun, Towsley Don, Gong Weibo. A Firewall Network System for Worm Defense in Enterprise Networks: [Umass ECE Technical Report TR-04-CSE-01]. 2004, 2: 1～13

10 Zou Cliff Changchun, Gong Weibo, Towsley D F. Code red worm propagation modeling and analysis under dynamic quarantine defense. In: ACM Conf. on Computer and Communications Security, 2003. 51～60