ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.18, No.2, February 2007, pp.412−421
DOI: 10.1360/jos180412

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

一种基于本地网络的蠕虫协同检测方法*

张新宇[1,2,3+], 卿斯汉[1,2,3], 李 琦[1,2,3], 李大治[1,3], 何朝辉[1,3]

[1](中国科学院 软件研究所 信息安全技术工程研究中心,北京 100080)

[2](北京中科安胜信息技术有限公司,北京 100086)

[3](中国科学院 研究生院,北京 100049)

A Coordinated Worm Detection Method Based on Local Nets

ZHANG Xin-Yu[1,2,3+], QING Si-Han[1,2,3], LI Qi[1,2,3], LI Da-Zhi[1,3], HE Zhao-Hui[1,3]

[1](Engineering Research Center for Information Security Technology, Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

[2](Beijing ZhongkeAnsheng Corporation of Information Technology, Beijing 100086, China)

[3](Graduate School, The Chinese Academy of Sciences, Beijing 100049, China)

+ Corresponding author: E-mail: zh.xinyu@tom.com

Zhang XY, Qing SH, Li Q, Li DZ, He ZH. A coordinated worm detection method based on local nets. *Journal of Software*, 2007,18(2):412−421. http://www.jos.org.cn/1000-9825/18/412.htm

Abstract: There are several global detection methods, but they do not apply to local net. A new cooperative approach to automatic detection of worms using local nets is presented in this paper, which is called CWDMLN (coordinated worm detection method based on local nets). This algorithm focuses on scanning worm characteristics in local nets and uses different methods to cope with different worm behaviors, including using honeypots to deceive worms. CWDMLN coordinates these methods to give graded alarms to notify worm attacks. The grades reflect reliability of alarms. Experimental results show that this approach is promising for it can quickly find worm intrusion in local nets and extract unknown worm signatures that can be used for IDS (intrusion detection system) or firewall to prevent more worm threats. This method can also contribute to global worm alarming by scaling.

Key words: Internet worm; honeypot; network attack; intrusion detection; propagation model

摘 要: 目前已有一些全球化的网络蠕虫监测方法,但这些方法并不能很好地适用于局域网.为此,提出一种使用本地网协同检测蠕虫的方法 CWDMLN(coordinated worm detection method based on local nets).CWDMLN 注重分析扫描蠕虫在本地网的行为,针对不同的行为特性使用不同的处理方法,如蜜罐诱捕.通过协同这些方法给出预警信息,以揭示蠕虫在本地网络中的活动情况.预警信息的级别反映报警信息可信度的高低.实验结果表明,该方法可以准确、快速地检测出入侵本地网络的扫描蠕虫,其抽取出的蠕虫行为模式可以为协同防御提供未知蠕虫特征.通过规模扩展,能够实施全球网络的蠕虫监控.

* Supported by the National Natural Science Foundation of China under Grant No.60573042 (国家自然科学基金); the National Grand Fundamental Research 973 Program of China under Grant No.G1999035802 (国家重点基础研究发展规划(973)); the Beijing Natural Science Foundation of China under Grant No.4052016 (北京市自然科学基金)

Received 2005-06-21; Accepted 2006-01-19

关键词: 网络蠕虫;蜜罐;网络攻击;入侵检测;传播模型
中图法分类号: TP393 文献标识码: A

近几年来,网络蠕虫已成为威胁因特网安全的首要因素.根据安全站点 SecurityPortal 的统计,在 2001 年的最具感染破坏能力的恶意代码排行榜上,蠕虫占据了 10 个席位中的 8 个[1].CERT/CC(Computer Emergency Response Team Coordination Center)的 2003 年度报告在关于入侵活动的总结中提到该年度两个最严重的入侵事件是由 Sobig.F 和 Slammer 蠕虫引起的[2].我国的互联网络也饱受蠕虫的侵袭[3].

蠕虫利用常见服务的安全漏洞或策略缺陷,通过网络进行传播.如果有漏洞的服务被安装在大量可公开访问的主机上,蠕虫就能自动入侵这些系统,造成大范围的感染.在发现和感染弱点系统阶段,蠕虫占用大量网络和系统资源,如果蠕虫负载具有攻击性,则还会造成窃取用户敏感数据、删除宝贵资源等无法挽回的后果.

因特网的开放环境为蠕虫的快速传播提供了便利条件.蠕虫最具威胁性的特点是传播速度很高.许多事件和研究[4,5]表明,一个工程良好的蠕虫能在几分钟内感染 90%以上的因特网弱点系统,严重扰乱网络正常秩序.这样快速的传播能力需要自动化的检测响应机制来进行有效防御.

随着人们对蠕虫的关注和研究,提出了一些蠕虫检测方法[6−8].这些方法都是针对大规模网络的算法,要求建立全球控制中心,监控大量的网络信息.这种全球监控方法主要有以下不足:① 要求合作者共享信息,而在现实中,许多机构出于种种原因不希望分享私密信息;② 有相当数量的共享信息传输会占用网络带宽,如果要建立专用网络,则需要一定的投资;③ 尽管从全局的角度能够实现早期检测,但这是以牺牲部分网络为代价的.

本文提出了一种使用本地网络实时协同检测蠕虫的方法 CWDMLN(coordinated worm detection method based on local nets),以避免大范围网络蠕虫检测的不足.对扫描蠕虫在本地网络中表现出的种种行为特性,CWDMLN 有针对性地使用不同的检测方法.通过协同处理给出预警信息,以达到及时通知本地用户蠕虫入侵的目的.实验证明,该方法适用于本地网络,可以准确、快速地检测出蠕虫入侵,其检测出的蠕虫传播模式可为协同防御提供未知蠕虫特征.

本文第 1 节讨论蠕虫检测的相关工作.第 2 节分析扫描蠕虫在本地网络的行为特性,给出本地网络蠕虫协同检测框架.第 3 节详细介绍 CWDMLN 算法.第 4 节给出实验结果和分析.第 5 节指出 CWDMLN 今后需要改进之处.

1 相关工作

在蠕虫研究中,常常以简单病毒传播模型为依据:

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

其中,$J(t)$是在 t 时刻被感染的系统个数;N 是易感系统总数;$\beta=\alpha/N$,α 是蠕虫感染率.当 $t=0$ 时,$J(0)$是初始状态时已被感染的系统个数.这一模型能够较好地描述扫描机制一致的蠕虫,特别是在传播的初期没有人工干预或网络拥塞可以忽略的情况下,能够刻画蠕虫的传播特性.

Cliff Zou 等人[6]使用上述模型研究蠕虫检测,通过选用卡尔曼滤波算法检测异常的扫描流量来发现蠕虫.

Jiang Wu 等人[9]利用离散模型研究蠕虫扫描特性,提出了基于受害主机数的检测算法:

$$n_{t+1}=n_t+[N-n_t]\left[1-\left(1-\frac{1}{T}\right)^{sn_t}\right] \tag{2}$$

其中,N 是易感系统总数;T 是蠕虫扫描选址空间大小;s 是平均扫描率;n_t 是第 t 个采样点被蠕虫感染的系统个数.

上述研究都是以全球互联网规模为参照的.如果考虑蠕虫爆发时所在网络的情况,特别是蠕虫在传播选址时优先选用本地网络,则本地网会很快被感染,如图 1 所示.

图 1 中的参数设定为:全球网 $N=360000$,$T=2^{32}$,$s=120$/s,$n_0=10$;本地网 $N=100$,$T=2^{16}$,$s=120$/s,$n_0=1$.图例说明:在同样的传播速度下,蠕虫需要 2 322s 的时间感染整个全球网的易感系统,但感染一个具体的 B 类网络中的所

有易感系统所花的时间却很少,不到 60s.因此,使用上述检测方法对蠕虫进行全球监控是以牺牲一些局域网络为代价的.

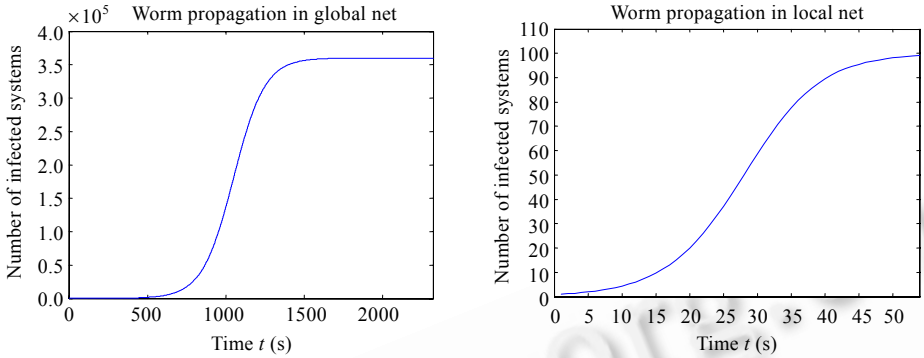


Fig.1 Comparison between worm propagation in global and local net

图 1 蠕虫全球传播与本地传播对比

利用上述模型检测蠕虫都假定蠕虫的传播速率恒定,并且弱点系统在 Internet 的分布是等密度的,这与实际情况不符.研究[10]表明,弱点系统的密度对传播是有影响的,蠕虫在有较多弱点系统的网络中传播更快.另外,初始感染系统数、系统修补措施以及网络延时等都对蠕虫传播有影响.

文献[6–9]提出的蠕虫检测算法属于异常检测方法.这些基于统计的方法需要采集和分析大量的数据来保证低误警率.

与上述异常检测法不同,Singh[11],Kim[12]和 Madhusudan[13]提出了特征抽取检测法.他们通过检测蠕虫负载的相似度和频度来提取蠕虫攻击特征,以发现蠕虫入侵.这些方法可以对付一些未知蠕虫,但却面临多态蠕虫的挑战.

针对大规模网络蠕虫检测的不足,Whyte 等人提出了两种具体的企业网蠕虫检测方法[14,15]:一种是利用蠕虫传播使用 IP 地址而很少用 DNS(domain name service)的特性来快速检测蠕虫活动;另一种是利用网络中异常的 ARP(address resolution protocol)流量来发现蠕虫.如果一些应用不使用 DNS,第 1 种检测方法则会导致较高的误警率.蠕虫传播途径可以分为:① 本地网到本地网;② 本地网到外网;③ 外网到本地网.第 2 种方法只能检测本地网到本地网的蠕虫传播.Antonatos 等人[16]使用不断变换的地址对抗在本地网络攻击固定地址(Hitlist)的蠕虫.但这种方法有一定的局限性,需要应用支持 DHCP(dynamic host configuration protocol).

除了采集网络流量检测蠕虫以外,人们还提出了基于主机的蠕虫检测方法.Malan 等人[17]通过分析对等的两个操作系统上的系统调用序列是否存在相似性来判断蠕虫活动情况.这类检测方法可以检测单系统上的蠕虫活动情况,但是无法预测大规模的蠕虫爆发.虽然主机检测能够提供更多的细节信息,但是需要对主机系统进行应用安装或系统改造,这势必会影响已有的应用.从管理和费用的角度来看,利用网络检测更具优势.

本文提出的 CWDMLN 算法是基于本地网络的,虽然采集的信息量相对较少,却能够及时发现本地网络中的蠕虫活动,保障本地网络的安全.CWDMLN 通过分析蠕虫在传播阶段的行为特点,可在蠕虫传播的早期向本地用户发出预警信息.这种利用蠕虫传播特性的检测手段比在流量负载中寻找特定的模式具有更高的抽象层次.蠕虫编写者很容易变换内容从而逃避特征抽取的检测方法,但却不容易改变传播特性.这种基于蠕虫行为的分析,使蠕虫活动模式具体化了,能够降低误警率.由于该算法不依赖于上述提到的模型,因此无须对扫描模式是否统一以及弱点系统密度分布等作出假设.该算法也不完全依赖蠕虫负载和传输内容,因此可以对抗多态蠕虫.通过提取蠕虫传播特征,CWDMLN 可以检测未知蠕虫.该算法在实施中利用了蜜罐构件,可以非常有效地发现蠕虫的可疑活动,提高检测的可信度和快速性.CWDMLN 是一种基于网络的检测方法,不会对主机系统造成任何影响.CWDMLN 使用多种检测手段,弥补了单一检测方法的不足.

2 检测框架

本节首先研究蠕虫传播的行为特性,然后给出利用本地网络进行蠕虫检测的框架.

2.1 蠕虫行为特性

本文研究的是自传播蠕虫,即利用常见服务的安全漏洞或策略缺陷,通过网络进行自传播的程序.这类蠕虫的攻击过程通常包括感染、传播和执行负载 3 个阶段.

在感染阶段,蠕虫已入侵到本地系统,它为了进行自治传播必须修改一些系统配置、运行程序,为后续阶段做准备,如生成多个线程以备快速探测新的目标之需、设置互斥标志防止系统被再次感染而影响传播速度等.通过这个阶段的工作,蠕虫将所入侵的系统变为一个新的感染源.这个感染源继续攻击过程以扩大感染范围.

在传播阶段,蠕虫要尽可能地发现可攻击的目标,利用漏洞获取系统权限,将自身从一个系统传播到另一个系统,从而入侵大量的目标.

蠕虫负载是指能够完成攻击者意图的代码.有些蠕虫包含负载,而有些蠕虫不包含负载.通过执行负载,蠕虫可以完成攻击任务,例如发动拒绝服务攻击、删除系统数据等.

从网络检测的角度来看,比较理想的时机是在蠕虫传播阶段,可以利用探测特性和自传播特性及早发现蠕虫活动.

在这一阶段,蠕虫会采用多种技术最大限度地发现可攻击目标.为了尽快发现攻击目标,在一定时间内会产生大量的网络探测报文.尽管蠕虫扫描策略有多种形式[9,18,19],如选择性随机扫描、可路由的扫描、分解扫描、DNS 扫描、完全扫描等,但在监视网活动时可以观测到如下的蠕虫网络活动模式:

(1) 一对多通信模式.为了感染更多的系统,蠕虫在利用诸如多线程等技术的同时向多个目标系统发送探测信息时,以及在发现了多个弱点系统以后进行自传播时,都会出现一对多的通信模式,如图 2 中的 A 节点所示.

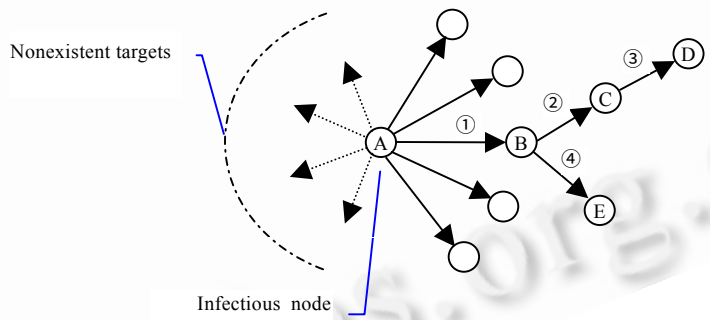


Fig.2 Action patterns of worms

图 2 蠕虫活动模式

(2) 相似的通信模式.蠕虫是自传播程序,包含用于探测和攻击弱点服务的特定模块.当蠕虫利用同样的模块探测和攻击时,就会出现这一特征.特别是当网络中所有的主机运行相同版本的服务时,即使蠕虫会变换使用不同的探测和漏洞利用模块,它们的数量也是有限的.在图 2 中,A 使用相同的目的端口探测其他目标系统、A 感染 B、B 感染 E 等,也利用相同的服务漏洞.如果一个节点使用相同的模式访问了多个节点,则称这种模式为花瓣通信模式.图 2 中的 A 节点就具有这种网络活动模式.

(3) 无效的连接访问.无效的连接访问是由于目标系统不存在或服务端口不存在所引起的.大多数的蠕虫选址策略含有随机因素,因此,蠕虫的网络活动中有相当数量的目标 IP 地址或服务不存在的报文.有针对性地分析这些报文就会加快蠕虫检测的速度.

(4) 链路通信模式.某节点只有先被感染,然后才能去感染其他节点.随着感染的扩散,会形成感染链路.如图 2 中的链路①-②-③和①-④.

2.2 检测框架

针对上面分析出的蠕虫传播的特点,给出相应的基于本地网络的蠕虫协同检测框架,如图 3 所示.

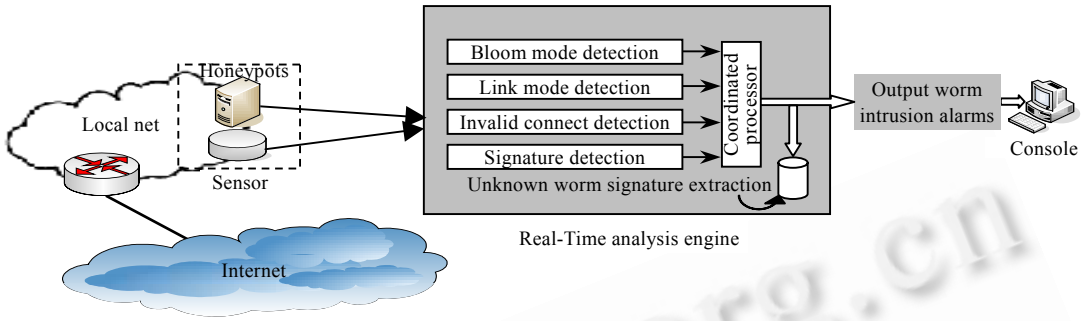


Fig.3 Cooperative worm detection framework based on local nets

图 3 本地网络蠕虫协同检测框架

为了监控本地网络活动,可选取一个探测点部署探测器,采集本地网络信息.例如,该点可以是交换机的镜像口.视资源情况,蜜罐可以部署在探测器上,也可以单独部署.蠕虫实时分析引擎对汇总的网络活动进行实时分析,给出可疑的蠕虫活动预警信息.控制台将预警信息进行可视化处理,描绘感染节点和传播途径.

蠕虫在探测阶段的网络活动见表 1.其中,"√"表示目标系统或服务存在,"×"表示目标系统或服务不存在,"–"表示没有应答响应.

Table 1 Features of worm probing/response

表 1 蠕虫探测/应答特征

Probing direction	Protocol	Target	Service	Response
Local node → Local node	TCP (transmission control protocol)	√	√ (TCP)	√
	UDP (user datagram protocol)		√ (UDP)	–
	TCP	√	× (TCP)	RST
	UDP		× (UDP)	ICMP-T3
	TCP	Honeypots	√ (TCP)	√
	UDP		× (UDP)	ICMP-T3
Local node → Remote node	TCP	√	√ (TCP)	√
	UDP		√ (UDP)	–
	TCP	√	× (TCP)	RST
	UDP		× (UDP)	ICMP-T3
	TCP/UDP	×	×	Router response: ICMP-T3
Remote node → Local node	TCP	√	√ (TCP)	√
	UDP		√ (UDP)	–
	TCP	√	× (TCP)	RST
	UDP		× (UDP)	ICMP-T3
	TCP	Honeypots	√ (TCP)	√
	UDP		× (UDP)	ICMP-T3

如果目标系统存在但不提供服务,则可监测到 RST[20]或 ICMP-T3[21]类型的应答报文.如果是目标系统不存在,则观测不到响应报文.通过在本地网络中部署蜜罐[22],占用本地网络未分配使用的 IP 地址来模拟特定类型的系统,就可以将对不存在地址的访问引入蜜罐.任何到蜜罐的访问都是可疑行为,因此,利用蜜罐采集到的信息其价值很高.蠕虫在攻击中由于选址问题会出现大量的无效连接.为此,通过分析本地网络无效连接的应答和对蜜罐的访问,可以迅速了解蠕虫的传播活动.

分析引擎还包括其他检测构件:花瓣通信模式检测、链路通信模式检测和特征检测.特征检测是指利用已知蠕虫模式来检测,这种检测方法可以利用公开信息对付已知的蠕虫.协同处理构件对上述检测方法进行权衡,给出预警信息,同时抽取蠕虫传播活动模式,该信息可以对抗未知蠕虫.如果将这些信息及时地传递给安全防护的其他功能部件,如防火墙或 IDS(intrusion detection system)等,则可以阻止蠕虫向内、外扩散,将蠕虫及时隔离.

3 CWDMLN 算法

定义 **1**. 用 S,P,D,T,Z 分别表示通信连接的源节点、目的服务、目的节点、事件发生时间和传输负载.通信模式 C 是五元组(S,P,D,T,Z)构成的关系.$c\in C$ 表示一个通信连接.

定义 **2**. 给定某节点 s',$bloom(s')_{p'}$=TRUE 表示节点 s'具有花瓣状通信模式的属性,即在给定的时段内访问了多个不同目的节点的相同服务 p'.T_B 为指定的阈值.

$$\text{if } \exists p'\in P\wedge\frac{|D_{s',p'}|}{\max T_{s',p'}-\min T_{s',p'}}\geq T_B \text{ ,then } bloom(s')_{p'}=\text{TRUE}.$$

where: $D_{s',p'}=\{c[D]|c[S]=s'\wedge c[P]=p'\}$;$T_{s',p'}=\{c[T]|c[S]=s'\wedge c[P]=p'\}$.

定义 **3**. 用 $Link(c_1,\ldots,c_n)_{p'}$=TRUE 表示 $c_1,\ldots,c_n$ 可构成一个通信链,且每个连接的目的服务都为 p'.T_L 为指定的阈值.$linknode(c_i[S])_{p'}$=TRUE 表示 $c_i[S]$分量是通信链 $Link(c_1,\ldots,c_n)_{p'}$中的一个节点.

$$\text{if } \exists c_i\in C\wedge\exists p'\in P\wedge c_{i+1}[T]>c_i[T]\wedge c_{i+1}[S]=c_i[D]\wedge c_i[P]=p'\wedge c_{i+1}[P]=p',$$
$$\text{then } Link(c_1,\ldots,c_n)_{p'}=\text{TRUE}\wedge linknode(c_i[S])_{p'}=\text{TRUE},i=1,\ldots,n(n\geq T_L).$$

定义 **4**. 用 $honeynode(s')$=TRUE 表示节点 s'是由蜜罐模拟的系统,$honeynode(s')_{p'}$=TRUE 表示模拟系统 s'提供的服务 p'也是模拟的.

$$\text{if } honeynode(s')=\text{TRUE, then } honeynode(s')_{p'}=\text{TRUE}.$$

定义 **5**. 用 $signature(c)$=TURE 表示一个通信连接 c 具有已知蠕虫行为特征模式,$signode(s')_{p'}$=TURE 表示 s'是 c 的源节点,该连接涉及的目的服务是 p',s'进行的这一网络活动是可疑的蠕虫活动.

$$\text{if } signature(c)=\text{TURE}\wedge c[S]=s'\wedge c[P]=p', \text{ then } signode(s')_{p'}=\text{TURE}.$$

定义 **6**. 用 w_1,w_2,w_3 和 w_4 表示权重,W 为指定的阈值,$infected(s)_p$=TURE 表示节点 s 已被感染,并试图利用服务 p 继续进行蠕虫传播.

$$\text{if } (w_1\times bloom(s)_p+w_2\times linknode(s)_p+w_3\times honeynode(s)_p+w_4\times signode(s)_p)>W, \text{ then } infected(s)_p=\text{TURE}.$$

CWDMLN 还对报文传输负载进行分析,抽取传输负载共性.如图 4 所示.

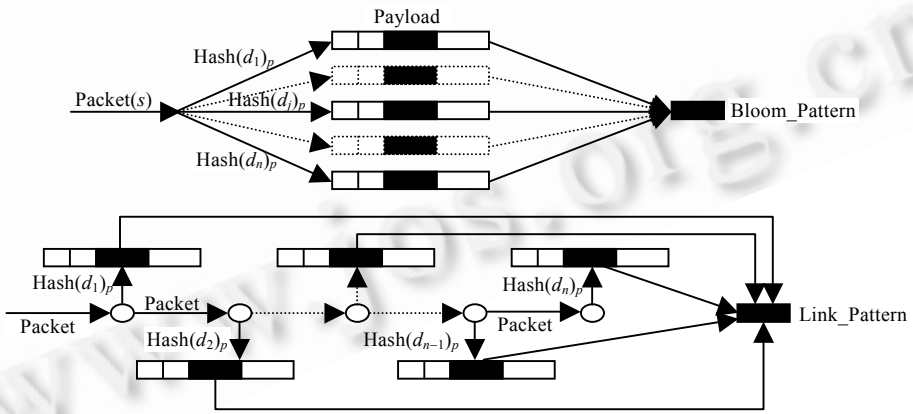


Fig.4 Analyze the payloads to extract the same pattern

图 4 分析传输负载以抽取相同模式

CWDMLN 算法描述如下:

```
Worm_detection(s,p,d,t,z)
{  //检测这一通信连接是否具有网络蠕虫活动特性
   if (w1×bloom(s)p+w2×linknode(s)p+w3×honeynode(s)p+w4×signode(s)p)>W {
     //检测到蠕虫活动,产生报警信息
     output(s,p,w);   //输出感染节点和网络活动,w 值显示该预警信息的可信度.
     if w1×bloom(s)p≥W/2 {
```

```
        Bloom_pattern=extract(s,p,d,t,z);   //抽取花瓣的活动模式
        if len(Bloom_pattern)>0
          output(bloom_dst,Bloom_pattern);   //输出花瓣模式和传输负载的共性
        else output(bloom_dst);   //输出花瓣模式的目的节点
      }
      if w2×linknode(s)p≥W/2 {
        Link_pattern=extract(s,p,d,t,z);   //抽取通信链的活动模式
        if len(Link_pattern)>0
          output(link,Link_pattern);   //输出通信链和传输负载的共性,链中的节点已被感染
        else output(link);   //输出通信链,链中的节点已被感染
      }
    }
  }
```

CWDMLN 对探测器采集到的报文以及到蜜罐访问的报文进行综合分析和判定,检测节点是否具有花瓣状通信模式、网络中是否存在链路通信模式、无效连接的情况如何,并利用已知蠕虫特征检测蠕虫活动.对上述方法检测出的信息,根据权值进行权衡,如果最终的信息值超过给定的阈值,则将蠕虫预警信息发送到控制台.该信息包含蠕虫感染节点、蠕虫活动所利用的服务、传播过程、传输负载特征等信息.这些信息可以为协同防御提供未知蠕虫的特征,用于检测新出现的蠕虫.

4 实验结果和分析

4.1 实验结果

部署一个局域网,网络地址为 192.168.0.0/16,真实系统的 IP 从 192.168.3.1 开始编址,本地网络中未分配使用的 IP 地址用蜜罐模拟.模拟系统开放所有的 TCP 服务,关闭所有的 UDP 服务.选用其中一台工作站(192.168.3.9)释放蠕虫样本.设置 w_1=0.5,w_2=0.6,w_3=0.6,w_4=0.3,W=0.5.在该实验中,只设置 Ramen 和 Nimda 的已知特征.实验结果见表 2.

Table 2 Results of the experiments

表 2 实验结果

Worm sample	Release time	Beginning time of network activity	Detection time	Worm probing/ propagation pattern	Address selection	Max reliability
Adore	15:07:40	15:07:43	15:07:43	53,111,515/TCP	61.11.×.×	0.5
Blaster	16:39:35	16:39:35	16:39:37	135/TCP	192.168.3.× 192.168.×.× 126.93.×.×	1.7
Cheese	15:21:45	15:21:45	15:21:47	10008/TCP	199.132.×.×	0.5
CodeGreen	16:11:45	16:11:45	16:11:47	80/TCP	×.×.×.×	0.5
Dvldr	16:07:10	16:07:10	16:07:12	445/TCP	4.196.×.× 67.200.×.×	0.5
Lion	15:12:25	15:13:25	15:13:28	53/TCP	153.227.×.×	0.5
Nimda	16:22:30	16:23:00	16:23:03	80,139/TCP	192.×.×.× 192.168.×.× ×.×.×.×	2.0
Ramen	15:26:45	15:26:46	15:26:48	21/TCP	109.98.×.×	0.8
SARS-Worm	15:37:20	15:37:22	15:37:23	137/TCP	44.169.×.× 97.198.×.× 192.168.×.× 223.249.×.×	1.1
Sasser.A	16:43:45	16:43:46	16:43:48	445/TCP	×.×.×.×	0.5
Sasser.B	16:50:35	16:50:35	16:50:37	445/TCP	×.×.×.×	0.5
Slapper	15:17:35	15:17:35	15:17:37	2001/UDP	38.239.×.×	0.5
Sorso	15:32:05	15:32:07	15:32:08	137/UDP	73.178.0.× 107.53.0.× 110.193.0.× 192.168.0.×	0.5

在表 2 中,① 由于一些蠕虫选址利用随机数产生,因此每次蠕虫释放所选择的地址有一定的差别;② 一些蠕虫从释放到产生真正的网络活动有一定的时间差;③ 符号“×”表示随着时间的变化,地址也发生变化.

4.2 结果分析

我们在网络中测试了 13 个蠕虫样本.从检测时间与蠕虫网络活动时间的对比来看,CWDMLN 算法可以在蠕虫显现出网络活动后的 0s~3s 内快速检测出蠕虫.

通过 CWDMLN 不仅可以了解蠕虫入侵本地网络,还可以了解蠕虫行为模式.实验前并不清楚蠕虫的网络活动模式,但实验结果能够告诉我们蠕虫探测或传播所使用的协议和端口.

从蠕虫的选址情况来看:一些蠕虫样本是向外网传播的(实验中的本地网为 192.168.0.0/16),如 Adore,cheese,Dvldr 等;一些蠕虫会根据本地网络地址进行传播,如 Nimda,SARS-Worm,Blaster,Sorso.无论本地网中的蠕虫是向外网传播还是在本地网内传播,CWDMLN 都可以检测出来.这是它协同多种检测手段的结果.除了在实验中的本地网释放蠕虫对 CWDMLN 进行测试以外,我们还将检测系统置于真实的网络环境中.CWDMNL 也可以检测从外网进入本地网络的蠕虫,这是因为 CWDMNL 在检测框架中考虑了这种情况的发生.

蠕虫选址方式所产生的 IP 地址不存在的情况越多,出现无效连接的情况就越多,访问蜜罐的概率随之增大,检测的速度也就越快.因此在本地网络配置时,如果本地网络的 IP 地址已分配满了,则可以考虑分成若干个子网,每个子网都留出一些地址给蜜罐模拟使用.这样,增加蠕虫碰到蜜罐的概率,可以及时发现蠕虫入侵.

现在,一些蠕虫更倾向于在受害系统的本地网络或临近网络进行传播.这是由于通常情况下,一个网络中同类型平台配置的概率较高,因此,采用这种传播方式可以大幅度提高感染的成功率.如果本地网络中的一台系统被感染,则蠕虫就很可能在这个网络快速传播开来.Nimda 蠕虫就是采用侧重于本地网络的选址方法,因此,CWDMLN 可以检测出它在本地网络中的爆发,且能利用多种方式检测出蠕虫的活动,可信度极高.通过传输负载分析还抽取出了它的传播特征,这与该蠕虫的分析报告相一致[23].在上述实验中,以本地网为依据进行选址的蠕虫,其相应的报警信息的可信度较高.这是由于这类蠕虫除了其他检测手段以外,肯定还能用蜜罐检测,因此检测方式比较多,协同处理后的可信度就高.

CWDMLN 利用报警信息的可信度来说明报警的准确性.如果利用多种检测手段发现了蠕虫活动,则该检测结果的可信度就比较高,表明误警的可能性则比较低.

蠕虫在传播的早期进行目标探测,这时,可以快速检测到感染系统的花瓣状通信模式.如果蠕虫侧重于本地传播,则通过蜜罐可以检测出蠕虫的可疑活动.随着传播的进行,又可以检测到链状感染路径.因此,报警信息的可信度不断提高,使得检测的准确度不断提升.控制台还可以给出网络感染情况和蠕虫传播链路信息.

从实验结果可以看出,CWDMLN 算法具有的优点是:① 利用本地网络数据进行检测,保障本地网络安全;② 针对蠕虫的传播阶段,可以在蠕虫传播的早期检测出蠕虫;③ 预警信息包含可信度,给出检测结果的准确度指数;④ 利用蜜罐模拟系统快速检测网络非法访问,节省资源开销.

5 结束语

本文在分析蠕虫传播特点的基础上提出了一种使用本地网协同检测蠕虫的算法 CWDMLN.该算法利用蠕虫的花瓣通信模式、无效连接以及在本地网中部署蜜罐等检测手段,协同给出蠕虫入侵的预警信息.实验证明该方法是可行的.

如果需要实施对全球网络的蠕虫检测,也可以通过对 CWDMLN 进行规模扩展来完成.在每个本地网络中部署相应的检测框架,设置多级控制中心,逐级汇总预警信息,就可以在监测本地网络蠕虫活动的同时监测全球网的蠕虫活动.

本检测方法今后还需不断完善,可以改进之处有:① 各个阈值和权值的选取.目前,算法的这些值是预先设定的,今后可以利用创建工作轮廓的自适应方法自动选取.② 蜜罐仿真度.目前的系统只是简单模拟系统的服务,如果要对抗智能黑客,则必须以本地网络环境为基准,有针对性地模拟特定的操作系统,包括网络协议栈特征.③ 蠕虫特征提取的灵活度.目前的 CWDMLN 只是进行粗粒度的特征提取,今后应朝着细粒度的方向发展.

致谢 在此,我们向对本文的工作给予支持和建议的文伟平博士表示感谢.

References:

[1] Lemos R. Fast-Spreading code is weapon of choice for net vandals. 2001. http://news.com.com/Year+of+the+Worm/2009-1001_3-254061.html?tag=st.rn#

[2] CERT. CERT® coordination center 2003 annual report. 2003. http://www.cert.org/annual_rpts/cert_rpt_03.html

[3] NCNIPC. Network security analysis report of national computer network intrusion protection center. 2004 (in Chinese). http://www.cert.org.cn/articles/statistic/common/2004060321713.shtml

[4] Moore D, Paxson V, Savage S, Shannon C, Staniford S, Weaver N. Inside the slammer worm. IEEE Security & Privacy, 2003,1(4): 33−39.

[5] Staniford S, Moore D, Paxson V, Weaver N. The top speed of flash worms. In: Paxson V, ed. Proc. of the 2004 ACM Workshop on Rapid Malcode. Washington: ACM Press, 2004. 33−42.

[6] Zou CC, Gao L, Gong W, Towsley D. Monitoring and early warning for Internet worms. In: Jajodia S, ed. Proc. of the 10th ACM Conf. on Computer and Communication Security. Washington: ACM Press, 2003. 190−199.

[7] Berk V, Bakos G, Morris R. Designing a framework for active worm detection on global networks. In: Cole JL, Wolthusen SD, eds. Proc. of the IEEE Int'l Workshop on Information Assurance. Darmstadt: IEEE Computer Society, 2003. 13−23.

[8] Berk VH, Gray RS, Bakos G. Using sensor networks and data fusion for early detection of active worms. In: Carapezza EM, eds. Proc. of the SPIE, Vol 5071. Orlando: SPIE, 2003. 92−104.

[9] Wu J, Vangala S, Gao L, Kwiat K. An effective architecture and algorithm for detecting worms with various scan techniques. In: Neuman C, ed. Proc. of the Symp. on Network and Distributed Systems Security (NDSS 2004). San Diego: Internet Society, 2004. 143−156.

[10] Rajab MA, Monrose F, Terzis A. On the effectiveness of distributed worm monitoring. In: McDaniel P, ed. Proc. of the 14th USENIX Security Symp. Baltimore: USENIX Association, 2005. 225−237.

[11] Singh S, Estan C, Varghese G, Savage S. Automated worm fingerprinting. In: Brewer E, Chen P, eds. Proc. of the 6th Symp. on Operating Systems Design and Implementation. San Francisco: USENIX Association, 2004. 45−60.

[12] Kim HA, Karp B. Autograph: toward automated, distributed worm signature detection. In: Blaze M, ed. Proc. of the 13th USENIX Security Symp. San Diego: USENIX Association, 2004. 271−286.

[13] Madhusudan B, Lockwood J. Design of a system for real-time worm detection. In: Lyles B, Watters A, eds. Proc. of the 12th Annual IEEE Symp. on High Performance Interconnects (Hot-I). Standford: IEEE Computer Society, 2004. 77−83.

[14] Whyte D, Kranakis E, Oorschot PV. Dns-Based detection of scanning worms in an enterprise network. In: Harder E, ed. Proc. of the 12th Annual Network and Distributed System Security Symp. (NDSS). San Diego: Internet Society, 2005. 181−195.

[15] Whyte D, Oorschot PV, Kranakis E. Arp-Based detection of scanning worms within an enterprise network. 2005. http://www.scs.carleton.ca/~kranakis/Papers/TR-05-02.pdf

[16] Antonatos S, Akritidis P, Markatos EP, Anagnostakis KG. Defending against Hitlist worms using network address space randomization. In: Keromytis AD, ed. Proc. of the 2005 ACM Workshop on Rapid Malcode. Fairfax: ACM Press, 2005. 30−40.

[17] Malan DJ, Smith MD. Host-Based detection of worms through peer-to-peer cooperation. In: Keromytis AD, ed. Proc. of the 2005 ACM Workshop on Rapid Malcode. Fairfax: ACM Press, 2005. 72−80.

[18] Weaver N, Paxson V, Staniford S, Cunningham R. A taxonomy of computer worms. In: Staniford S, ed. Proc. of the 2003 ACM Workshop on Rapid Malcode. Washington: ACM Press, 2003. 11−18.

[19] Wen WP, Qing SH, Jiang JC, Wang YJ. Research and development of Internet worms. Journal of Software, 2004,15(8):1208−1219 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/15/1208.htm

[20] DARPA INTERNET PROGRAM. Transmission control protocol. 1981. http://rfc.net/rfc793.ps

[21] DARPA INTERNET PROGRAM. Internet control message protocol. 1981. http://rfc.net/rfc792.ps

[22] Spitzner L. Honeypots: Definitions and value of honeypots. 2003. http://www.tracking-hackers.com/papers/honeypots.html

[23] Mackie A, Roculan J, Russell R, Velzen MV. Nimda worm analysis. 2001. http://www.dia.unisa.it/professori/ads/corso-security/www/CORSO-0102/NIMDA/link_locali/010921-Analysis-Nimda-v2.pdf

附中文参考文献:

[3] 国家计算机网络入侵防范中心(NCNIPC).国家计算机网络入侵防范中心网络安全分析报告.2004. http://www.cert.org.cn/articles/statistic/common/2004060321713.shtml

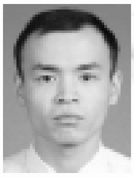
[19] 文伟平,卿斯汉,蒋建春,王业君.网络蠕虫研究进展.软件学报,2004,15(8):1208−1219. http://www.jos.org.cn/1000-9825/15/1208.htm

张新宇(1968－),女,湖南宁乡人,博士生,工程师,主要研究领域为网络信息安全.

李大治(1976－),男,工程师,主要研究领域为网络信息安全.

卿斯汉(1939－),男,研究员,博士生导师,CCF 高级会员,主要研究领域为信息系统安全理论和技术.

何朝辉(1980－),男,工程师,主要研究领域为网络信息安全.



李琦(1979－),男,工程师,主要研究领域为网络信息安全.

△△