一种基于信用的综合蠕虫检测算法

杨宏宇，米耘锋

(中国民航大学计算机科学与技术学院 天津 东丽区 300300)

【摘要】提出一种基于信用的综合蠕虫检测算法(CBCWD)。首先，通过检测对比实验对连接成功概率参数进行优化设定；其次，在对基于信用的连接率限制算法响应策略进行了重新设计，以此为基础对该算法的响应处理过程进行了改进。在上述研究基础上，设计一种基于信用的综合蠕虫检测算法。该算法通过监测网络流量信息，对监测事件执行响应策略，通过攻击条件判定和综合分析得出检测结果。采用NUST 2011开放数据集进行5种不同频率的攻击检测对比实验，实验结果表明该算法具有更高的检测率和更低的误报率。

关 键 词 信用; 数据集; 检测; 连续假设检验; 蠕虫扫描

中图分类号 TP309; TP393.08 文献标志码 A doi:10.3969/j.issn.1001-0548.2013.04.008

A Credit-Based Comprehensive Worm Detection Algorithm

YANG Hong-yu and MI Yun-feng

(School of Computer Science and Technology, Civil Aviation University of China Dongli Tianjin 300300)

Abstract This paper presents a new comprehensive worm detection algorithm based on credit. Firstly, the parameters of connection success probability are optimized and set through detection contrast experiments. Secondly, the response strategy of the credit-based connection rate limiting algorithm is redesigned, and consequently the event response process of the algorithm is improved. On the basis of the above study, the new credit-based comprehensive worm detection (CBCWD) algorithm is designed to monitor network traffics, execute response strategy for monitoring events, and give the detection results through attack condition judge and comprehensive analysis. Five different frequencies attacks detection experiments on the NUST 2011 open dataset were conducted. The experiment results demonstrate that proposed algorithm has a higher detection rate and lower false positive rate.

Key words credit; dataset; detection; sequential hypothesis testing; worm scan

随着互联网的迅速发展，电子商务、视频音频、社交网络等各种网络应用已经越来越广泛，人们对网络的安全性需求越来越迫切。与此同时，网络流量在近年来发生了巨大的变化，P2P流量占总流量的比重大幅增加，未来仍将继续增加[1]。由于P2P流量的许多特性和蠕虫攻击的特性十分相似，导致原来很多成功的异常检测方法性能下降，这对现在的异常检测提出巨大的挑战。另一方面，各种网络攻击手段更加具有威胁性、隐蔽性。扫描攻击采用一对多的攻击拓扑形式，寻找主机漏洞和基于漏洞进行新一轮攻击从而获取利益为目标，对网络造成极大的破坏[2-3]。

1 相关工作

目前主流的异常检测方法有极大熵估计方法(MAXENT)[4]、阈值随机浮动算法(TRW)[5]、连接率限制算法(rate limiting)[6-7]、基于包的网络异常检测(NETAD)[8]和基于信用的阈值随机浮动算法(TRW-CB)[9]。

文献[4]提出极大熵估计方法(MAXENT)。该方法对正常流量采用极大熵估计技术计算基线值，通过计算观测的网络流量的相对熵值，与基线值进行对比，判断观测的流量是否为恶意流量。该方法具

收稿日期：2011 – 09 – 23；2011 – 12 – 06

基金项目：国家自然科学基金(60776807, 61179045)；国家863计划重点课题(2006AA12A106)；天津市科技计划重点项目(09JCZDJC16800)；中国民航科技基金(MHRD201009, MHRD201205)；中央高校基本科研业务费专项资金(ZXH2009A006, ZXH2010D009)

作者简介：杨宏宇(1969 –)，男，博士，教授，主要从事网络与信息安全方面的研究.

有较高的检测率和较低的误报率，但缺点是检测结果延迟偏大[10]。

文献[5]提出阈值随机浮动算法(TRW)。该算法采用基于连续假设检验的方法，根据每次连接的结果进行相应的计算，再与可以灵活设定的阈值进行比较从而判断发起连接方是否是病毒攻击。该方法的缺点是判断一个系统是否被感染有明显的延迟，而当判断结果出现时，蠕虫可能已经进行扩散和新的攻击。

文献[6-7]提出连接率限制算法(rate limiting)。该算法首先监测连接，如果是已经有记录的连接则允许直接连接，如果是一次新的连接且发起连接数目很大，则将该连接放入缓冲队列，延时发起。该算法可以对病毒扩散起到减弱作用，但本身的性能并不突出[10]。

文献[8]提出基于包的网络异常检测(NETAD)方法。该方法构建的NETAD模型由IP包头的前48个字节构成，对应模型的48个属性，并定义计算公式，计算出每一个包的积分(score)，从而进行判断。该方法易于实现，简化分析包结构也能得到较好的性能。其不足是由于只考虑很少的负载信息，基本无法检测负载中含有攻击信息的病毒攻击。

文献[9]提出一种基于信用的连接率阈值浮动(threshold random walk with credit-based rate limiting，TRW-CB)算法，是采用连续假设检验[5]和连接率限制[6-7]相结合的算法，其融合了这两种方法的特点从而达到更好的性能，但缺点是不能适应当前网络流量的特征。

本文提出一种基于信用的综合蠕虫检测(credit-based comprehensive worm detection ,CBCWD)算法。通过检测对比实验对基于连续假设检验的检测算法参数θ_0、θ_1 进行优化设定，对基于信用的连接率限制算法的事件响应策略进行了改进，以此为依据改进基于信用的连接率限制算法。在上述研究基础上，设计并实现一种基于信用的综合蠕虫检测算法，实现了对蠕虫攻击的有效检测。

2 基于连续假设检验的蠕虫检测

2.1 连续假设检验

文献[5,9]给出一种连续假设检验方法。当本地主机l向目的地址d初始化一个首次连接请求(first-contact connection)，则将结果分为“成功”或“失败”。如果请求是一TCP SYN包，在时间超时之前收到d的SYN-ACK则为成功。如果是一个UDP包，在时间超时之前收到任何从d回复的UDP包都为成功。定义：

$$Y_i=\begin{cases}0 & \text{连接成功}\\ 1 & \text{连接失败}\end{cases} \tag{1}$$

式中，随机变量Y_i 代表主机l的第i次连接请求。

定义H_1为假设主机l正在进行扫描攻击(被蠕虫感染)，H_0为假设主机没有进行扫描攻击。假设在条件H_j下，随机变量$Y_i|H_j$ i=1,2,… 是独立同分布。在此假设下，任意两次连接尝试将有相同的连接成功率，它们连接成功的概率彼此互不相关。因此，可采用Bernoulli随机变量Y_i表示分布：

$$\begin{cases}P\left[Y_{\mathrm{i}}=0\,|\,H_0\right]=\theta_0, \quad P\left[Y_{\mathrm{i}}=1\,|\,H_0\right]=1-\theta_0\\ P\left[Y_{\mathrm{i}}=0\,|\,H_1\right]=\theta_1, \quad P\left[Y_{\mathrm{i}}=1\,|\,H_1\right]=1-\theta_1\end{cases} \tag{2}$$

一次由正常主机发起的连接成功率很可能高于由扫描攻击发起的连接，即$\theta_0>\theta_1$。

选择两种假设并对比其可能性，将在每一种假设下产生一系列观测事件 $Y_n\equiv(Y_1,Y_2,\cdots,Y_n)$)，这种比值关系为：

$$\Lambda(Y_n)\equiv\frac{P\left[Y_n\,|\,H_1\right]}{P\left[Y_n\,|\,H_0\right]} \tag{3}$$

式(3)中分子是在假设H_1下的一系列事件Y_n的能性，分母是在假设H_0下的情况。

由独立同分布的假设，用比值表示独立事件的可能性关系为：

$$\Lambda(Y_n)\equiv\prod_{i=1}^{n}\frac{P\left[Y_i\,|\,H_1\right]}{P\left[Y_i\,|\,H_0\right]} \tag{4}$$

以第i次观测结果，即式(5)的$\Phi(Y_i)$作为$\Lambda(Y_n)$公式的变换：

$$\Phi(Y_i)\equiv\frac{P\left[Y_i\,|\,H_1\right]}{P\left[Y_i\,|\,H_0\right]}=\begin{cases}\dfrac{\theta_1}{\theta_0}, Y_i=0 & \text{成功}\\ \dfrac{1-\theta_1}{1-\theta_0}, Y_i=1 & \text{失败}\end{cases} \tag{5}$$

综合式(4)和式(5)，可迭代计算每次观测结果：

$$\Lambda(Y_n)=\prod_{i=1}^{n}\Phi(Y_i)=\Lambda(Y_{n-1})\Phi(Y_n) \tag{6}$$

式中，$\Lambda(Y_0)$=1。

每次连接请求事件得知结果，计算新的$\Lambda(Y_n)$。$\Lambda(Y_n)$与上限阈值η比较，大于η时，则接受假设H_1。

2.2 蠕虫检测算法

蠕虫是一种可以自动从主机向主机传播的病毒。蠕虫可以在被感染的主机上产生一系列的地址列表(地址列表是连续的或伪随机的产生)，然后探测和连接它们发起攻击。

该蠕虫检测算法的思路是针对大部分蠕虫攻击和首次接触连接请求，响应该连接请求的连接事件相关的特点，监测这些事件，并根据监测结果做出判定。

当在一个蠕虫检测系统中运行这个检测算法时，必须监测每一个主机各自的状态信息。因此，每个主机l对应一个Λ_l 。同时为每一个主机记录该主机前续连接主机 (previously contacted hosts，PCH) 集合。最后，每个主机l检测中状态信息等记录在FCC(first-contact connection)队列结构中，FCC队列结构如图1所示。

```
enum
st{PENDING,SUCCESS,FAILURE};
struct FCC_Queue_Element{
st Status;
boolean isMalicious;
time ConnectBegin;
ip4_add DestinationAdd;
}
```

图1　首次连接请求(FCC)队列元素的结构

文献[9]给出蠕虫检测算法的实现。该算法由3个事件触发，其描述如下：

1) 当蠕虫检测系统观测到一个由本地主机l发送的包(TCP SYN或UDP)，则检查它的目的地址d是否在l的前续连接主机集合中。若不在，则将d加入前续连接主机集合并在FCC队列的最后增加一条以d为目的地址且状态为待定(PENDING)的记录。

2) 当一个包到达本地主机l且其源地址也是l的FCC队列的一条记录的目的地址时，该包被认为是首次连接请求的响应，同时更新FCC记录的状态。如果这个包不是一个TCP RST包，则FCC记录的状态设置为成功(SUCCESS)。

3) 只要在FCC队列中的记录的状态是待定，并且在队列的时间超过连接超时周期，则超时发生，同时该记录的状态改为失败(FAILURE)。

当上述事件导致FCC队列的记录状态不为待定，则从FCC队列中删除该记录，通过式(6)计算得到Λ_l，并与η进行比较。如果$\Lambda_l \geqslant \eta$，则停止对主机l的检验并确认l被感染或攻击。只要$\Lambda_l<\eta$，FCC队列的记录状态不同于待定且队列不为空，则继续删除该记录。

2.3　连接成功概率参数设定

在基于连续假设检验的蠕虫检测算法中，首先根据经验预先设定连接成功概率参数θ_0、θ_1。本文根据检测实验结果对θ_0、θ_1值进行了设定，如表1所示。在实验中采用100SYN/1 s攻击样本数据，按照不同的θ_0、θ_1值进行攻击检测，根据最优检测结果设定θ_0=0.6，θ_1=0.2。

表1　不同参数值的检测对比实验

参数值	检测率/(%)	误报率/(%)
θ_0=0.9　θ_1=0.1	99.983 2	0.337 7
θ_0=0.8　θ_1=0.2	99.983 2	0.268 5
θ_0=0.7　θ_1=0.1	99.983 2	0.197 7
θ_0=0.6　θ_1=0.2(本文采用)	99.983 2	0.180 5

在该实验中采用的其他4种攻击样本数据分别为1SYN/10 s、1SYN/1 s、10SYN/1 s和1 000SYN/1 s。实验结果显示，θ_0=0.6且θ_1=0.2时，算法对攻击样本的检测误报率最低。

针对实验结果和连接成功概率参数θ_0、θ_1的设定值，通过分析可知：一方面，P2P流量在目前网络流量中已经占到40%～70%[11]，故无法忽略P2P流量的影响；另一方面，根据目前局域网中P2P流量较大的特点，资源主机有可能主动拒绝连接。如在局域网中进行P2P高速下载时，需要在服务器端保留资源IP地址且该IP地址的连接应具有较高的成功率，但是实际连接可能由于下载工具的原因而被资源方主动拒绝。上述现象导致正常网络请求成功连接的概率更低，因此，θ_0小于文献[5]和文献[9]的设定值。

3　基于信用的综合蠕虫检测算法

本文提出一种基于连续假设检验的检测算法和基于信用的连接率限制改进算法的综合蠕虫检测算法，即基于信用的综合蠕虫检测算法。该算法是在连续假设检验基础上，结合基于信用的连接率限制改进算法对网络流量进行蠕虫检测判定。

3.1　基于信用的连接率限制算法

文献[9]提出基于信用的连接率限制算法。但是其描述较简单。根据该算法思路，本文对信用的定义给出更明确的解释。信用就是体现网络对主机的信任程度，是网络主机的连接凭证，具有信用的主机才可以发起连接，成功连接可以得到更多的信用，失败连接将会降低信用，没有信用的主机不能在网络中发起连接。

基于信用的连接率限制算法为每一个主机l赋予一个信任值C_l，并根据一系列监测事件，改变C_l值。当事件使C_l <1时，则判定主机受到攻击；否则继续事件监测。

文献[9]的算法对事件的响应策略如表2所示。该算法根据监测事件计算C_l值。

表2 算法的事件响应策略

事件	改变C_l
初始值	$C_l = 10$
由l发起FCC	$C_l = C_l - 1$
FCC连接成功	$C_l = C_l + 2$
每秒	$C_l = \max(10, 2/3\ C_l)$如果$C_l > 10$
$C_l = 0$	$C_l = 1$ 如果$C_l = 0$持续4 s

3.1 基于信用的连接率限制改进算法

本文对基于信用的连接率限制算法进行了改进，提出一种基于信用的连接率限制改进算法。该算法与连续假设检验的区别在于它对信任值的计算不需要新的证据，所以响应速度更快。该算法将上述两种方法融合，使检测更准确。

基于信用的连接率限制改进算法的判定条件是当主机l被感染并对其他节点发动攻击时，C_l则相应减小。当C_l小于1时，则判定主机l被感染。

基于信用的连接率限制改进算法的事件响应策略设计如表3所示。在表3中，每一个局域网的主机l都维持一个相应的工作区。根据某个事件条件，改进算法执行响应策略并更新信任值C_l，直至满足判定攻击条件时，则判定攻击；否则继续事件监测。

表3 本文算法的事件响应策略

事件	改变C_l
初始值	$C_l = 15$
由l发起FCC	$C_l = C_l - 1$
FCC连接成功	$C_l = C_l + 8$
每秒	$C_l = \max(5, 3/4\ C_l)$如果$C_l > 5$
$C_l = 0$	$C_l = 1$ 如果$C_l = 0$持续4 s

在对算法响应策略重新设计的基础上，对算法过程进行了改进：

1) 主机l的C_l初始值设定为15，用于发起首次连接请求。过大或过小的C_l值都会降低算法性能(降低检测率和提高误报率)，C_l初始值较小，主机可能因发起若干次连接后C_l=0，被判定为攻击；若C_l初始值较大，可能为蠕虫发起大量攻击连接提供条件。

2) 当发出一个连接请求，则连接发起方主机的C_l减1；若连接成功建立，则C_l=C_l+8。蠕虫攻击连接成功率往往低于正常流量，所以会因没有C_l累加，导致C_l=0而被检测为蠕虫攻击；正常流量则因有C_l累加而继续正常网络连接。当连接成功建立时，C_l=C_l+8。所以，改进算法符合当前网络流量的特征。如P2P流量和蠕虫攻击具有相似特性，即在短时间内发起大量连接请求，但P2P流量的连接成功率较高，因此改进算法可以使其C_l增加，从而允许其在短时间内发起大量连接请求。

3) 当正常主机C_l累加值较高且被蠕虫感染后，该C_l值可能被蠕虫用于发起攻击连接请求，为避免以上的情况设计每秒事件，即每秒对信任值较高的主机进行一定比例的扣除，这种削弱不影响其正常的网络连接。

4) 当发生某事件并使C_l=0时，则判定为攻击。同时，延迟(每4 s)对该主机的C_l加1补偿操作，允许该主机发起一次连接请求。

3.2 基于信用的综合蠕虫检测算法

基于信用的综合蠕虫检测算法的执行过程从初始化开始，监测网络流量信息，对监测事件执行响应策略并进行攻击条件判定，若满足攻击判定条件，则判定为攻击；否则判定为正常并继续进行监测。

基于信用的综合蠕虫检测算法操作流程如图2所示。

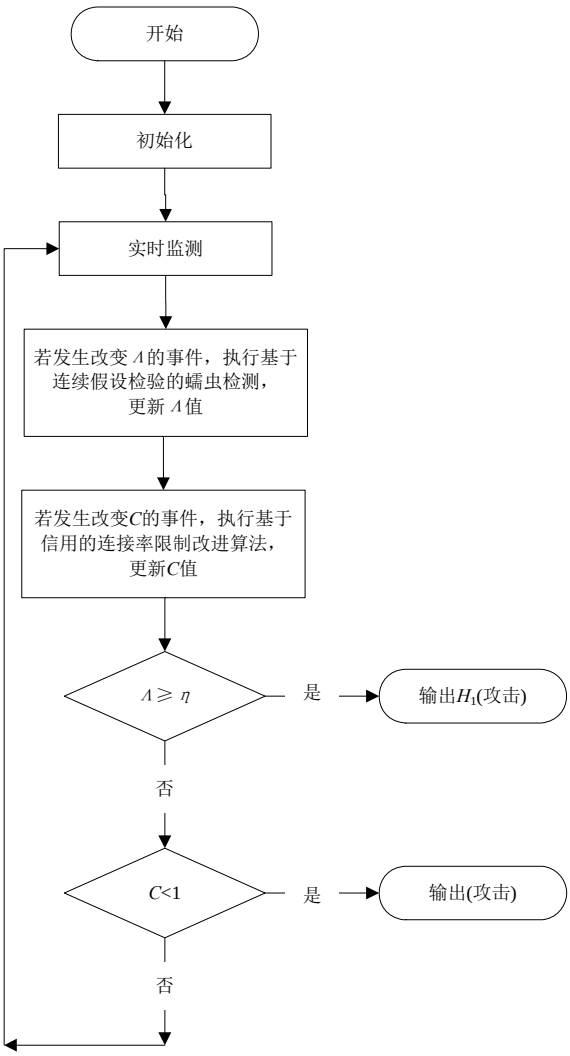

图2 算法操作流程

该算法描述如下：

1) 初始化阶段：初始化各项参数。

2) 监测阶段：若发生改变Λ的事件，则执行基于连续假设检验的蠕虫检测，更新Λ值；若发生改变C的事件，则执行基于信用的连接率限制改进算法并根据表3的策略执行响应操作，更新C值。

3) 判定阶段：如果Λ值满足攻击判定条件，则判定为蠕虫攻击；如果C值满足攻击判定条件，则判定为蠕虫攻击；否则，进入步骤2)，继续监测。

4 实验与结果

为验证本文提出的蠕虫检测算法的检测性能，采用公开、标记的NUST WisNet数据集[12]进行检测实验。

4.1 实验处理

由于DARPA评估数据集不能反映目前主流网络的流量特征[13-14]，而NUST WisNet数据集能够反应当前网络流量的特征，并具有真实性、标记性(标出正常流量和攻击流量)和开放性的显著特点，所以本文研究采用NUST WisNet实验室公开的2011版Benign Home Dataset和Attack Dataset进行攻击检测实验，以验证本文的检测方法性能。

该检测实验主要关注TCP连接。TCP通过“三次握手”建立网络连接，请求连接方通过发送SYN数据包请求同步，被请求方发送SYN-ACK数据包进行确认，最后连接方再次发送ACK数据包确认，双方才能正式建立连接；如果连接失败，则被请求方通过RST数据包表示异常终止一个TCP连接。

蠕虫攻击可利用TCP连接特性，发送大量SYN数据包并不建立连接，最终耗尽被连接主机的系统资源并在很短的时间内可以使被攻击主机失去连接和服务功能。

本文实验主要从攻击数据集中提取5组不同频率的攻击样本进行对比实验。实验的重要参数设定如表4所示。在实验中，将连接成功概率参数θ_0、θ_1分别设定为0.6和0.2，上限阈值η设定为100 000。

表4　重要参数设定表

重要参数	设定值
θ_0	0.6
θ_1	0.2
η	100 000

4.2 实验结果

为了证明本文算法的有效性，在实验中对TRW算法、TRW-CB算法和本文的CBCWD算法进行了攻击检测，实验结果如表5所示。

表5　多种攻击频率下的实验结果对比

攻击频率	TRW		TRW-CB		CBCWD(本文算法)	
(次数攻击名称/时间)	检测率/(%)	误报率/(%)	检测率/(%)	误报率/(%)	检测率/(%)	误报率/(%)
1SYN/10 s	56.666 7	0.946 6	66.666 7	0.313 3	83.333 3	0.180 5
1SYN/1 s	89.657 6	0.946 6	96.644 3	0.313 3	98.322 1	0.180 5
10SYN/1 s	97.922 7	0.946 6	99.664 9	0.313 3	99.832 4	0.180 5
100SYN/1 s	99.397 3	0.946 6	99.966 5	0.313 3	99.983 2	0.180 5
1 000SYN/1 s	99.931 7	0.946 6	99.996 7	0.313 3	99.998 3	0.180 5

表5中的检测结果说明，本文提出的基于信用的综合蠕虫检测(CBCWD)算法对蠕虫攻击的检测率更高，且对正常流量的误报率更低。

检测率是正确检测出的攻击样本数与攻击样本总数的比值。表5中的检测率是通过对5种攻击样本数据集的检测结果统计而得出。由于实验中攻击样本数据集为5种不同频率的攻击样本数据，因此检测率不同。误报率是被错误判断为攻击的正常样本数与正常样本总数的比值。表5中的误报率是通过对一个正常样本数据集的检测误报结果统计而得出。由于检测实验所采用的正常样本数据集是唯一的，因此表5中所有算法的检测误报率是相同的。

从表5可见，与TRW算法、TRW-CB算法相比，本文提出的CBCWD算法对5种不同频率的SYN攻击具有更高的检测率，CBCWD算法在对1SYN/10 s攻击的检测效果率尤为明显。这说明CBCWD算法在对某些蠕虫变种(如为躲避检测系统而降低扫描速率的蠕虫)具有更好的检测效果。

5 结 束 语

本文对基于连续假设检验的检测算法关键参数进行重新设定，并改进基于信用的连接率限制算法响应策略。在此基础上，设计并实现一种基于信用的综合蠕虫检测算法。通过NUST2011数据集检测实验，表明本文提出的算法对低频率蠕虫攻击有更明显的检测效果，且误报率较低。

由于目前攻击样本种类的局限，还不能进行更多攻击样本的检测对比实验，未来工作的重点是进

行该算法的检测实验并进一步完善其的响应策略。

参 考 文 献

[1] HAQ I U, ALI S, KHAN H, et al. What is the impact of P2P traffic on anomaly detection[C]//Proceedings of the 13th International Symposium on Recent Advances in Intrusion Detection. Ottawa, Canada: [s.n.], 2010: 1-17.

[2] ZHOU C V, LECKIE C, KARUNASEKERA S. A survey of coordinated attacks and collaborative intrusion detection[J]. Computers Security, 2010, 29(1): 124-140.

[3] STANIFORD S, PAXSON V, WEAVER N. How to own the internet in your spare time[C]//Proceedings of the 11th USENIX Symposium on Security. San Francisco, USA: [s.n.], 2002.

[4] GU Y, MCCULLUM A, TOWSLEY D. Detecting anomalies in network traffic using maximum entropy estimation[C]// Proceedings of the ACM/USENIX Conference on Internet Measurement. Berkeley, USA: [s.n.], 2005.

[5] JUNG J, PAXSON V, BERGER A W, et al. Fast portscan detection using sequential hypothesis testing[C]// Proceedings of the IEEE Symposium on Security and Privacy. Oakland, USA: IEEE, 2004: 211-225.

[6] TWYCROSS J, WILLIAMSON M M. Implementing and testing a virus throttle[C]//Proceedings of the 12th USENIX Symposium on Security. Washington, USA: [s.n.], 2003.

[7] WILLIAMSON M M. Throttling viruses: Restricting propagation to defeat malicious mobile code[C]// Proceedings of the 18th Annual Computer Security Applications Conference. Las Vegas, USA: [s.n.], 2002: 61-68.

[8] MAHONEY M V, CHAN P K. Network traffic anomaly detection based on packet bytes[C]//Proceedings of the ACM Symposium on Applied Computing. Florida, USA: [s.n.], 2003.

[9] SCHECHTER S E, JUNG J, BERGER A W. Fast detection of scanning worm infections[C]//Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection. Sophia Antipolis, France: [s.n.], 2004: 59-81.

[10] ASHFAQ A B, ROBERT M J, MUMTAZ A, et al. A comparative evaluation of anomaly detectors under portscan attacks[C]//Proceedings of the 11th International Symposium on Recent Advances in Intrusion Detection. Cambridge, USA: [s.n.], 2008: 351-371.

[11] SCHULZE H, MOCHALSKI K. Ipoque internet study report 2008/2009[EB/OL]. [2011-01-05]. http://www.ipoque.com/resources/internet-studies/internet-study-2008_2009.

[12] KHAYAM A. The dataset collected by WiSNet in NUST[EB/OL]. [2011-09-05]. http://www.wisnet.seecs.nust.edu.pk/downloads.php.

[13] MCHUGH J. The 1998 lincoln laboratory IDS evaluation[C]//Proceedings of the Third International Symposium on Recent Advances in Intrusion Detection. Toulouse, France: [s.n.], 2000: 145-161.

[14] MAHONEY M V, CHAN P K. An analysis of the 1999 DARPA/Lincoln laboratory evaluation data for network anomaly detection[C]//Proceedings of the 6th International Symposium on Recent Advances in Intrusion Detection. Pittsburgh, USA: [s.n.], 2003: 220-237.

编 辑 漆 蓉