第 33 卷 第 3 期 计 算 机 工 程 2007 年 2 月
Vol.33 No.3 Computer Engineering February 2007
·安全技术· 文章编号：1000—3428(2007)03—0160—03 文献标识码：A 中图分类号：TP309.1

基于进程行为的入侵检测系统的设计

吴 玉，陆晓君
(安徽大学交通分校，合肥 230051)

摘 要：在基于多层感知器的神经网络分类器和基于概率预测的贝叶斯分类器的基础上，给出针对描述系统进程行为的系统调用短序列进行分类的方法，用以识别被监控系统关键程序的执行过程中的系统调用是否正常。并研究系统中多个系统关键程序的运行监控问题，提出了一个基于进程行为分类的入侵检测系统原型。该系统原型能够根据系统配置，同时对系统中的多个系统关键程序的执行进行监控。
关键词：进程行为；神经网络；贝叶斯分类器；入侵检测系统

Design of Intrusion Detection Systems Based on Process Behavior

WU Yu, LU Xiaojun
(Communication School of Anhui University, Hefei 230051)

【Abstract】 Based on neural network classifier of multilayer perceptron and Bayesian classifier of probability prediction, the classification method is given of system call short sequence of description system process behavior, to identify whether the system call is normal or not in the performance process of monitor system key programs. The running and monitor problems of many system key programs in the system are researched and IDS prototype is put forward based on process behavior classifier. Based on system configuration, the system prototype can monitor the performance of many system key programs in the system.
【Key words】 Process behavior; Neural network; Bayesian classifier; IDS

任何一个计算机系统、软件应用等都存在一些安全性缺陷、漏洞或弱点。由于系统中的特权程序(例如 UNIX 环境下以 Root 权限运行的程序)能够获得系统控制权限，如果这些程序在设计和编程过程中遗留下来的错误和缺陷被入侵者利用，就可能使入侵者获得整个系统的控制权，这必然严重威胁系统的安全。目前已有许多这类的入侵攻击手段和工具。如果我们建立的入侵检测系统能够通过监控系统特权程序的执行，就能够有效地抵御或检测这些针对系统特权程序的攻击行为。

系统关键程序的执行，可通过程序执行过程中所使用的系统调用所组成的序列(称为程序执行迹)来描述。因为一个程序的代码具有相对的稳定性，所以程序的正常执行迹基本一致，但程序不正常执行(程序遭受攻击、被篡改或执行不正常执行的程序分枝等)时，产生的系统调用序列与程序正常执行时有很大的差别，就好像在执行其他程序一样。一个程序的正常行为可以由其执行迹中的局部模式来描述，因此，可以通过查表判别进程执行迹中的短序列是否为模式集中的序列，然后根据定长时间段内异常性序列出现的频度，来确定系统是否受到了入侵。但是，如果程序的模式表比较大，这种查表的方法必然不能满足入侵检测的实时性要求。

基于多层感知器的神经网络分类器和基于概率预测的贝叶斯分类器[1]的基础上，针对描述系统进程行为的系统调用短序列进行分类，用以识别被监控系统关键程序的执行过程中的系统调用是否正常。程序执行迹只需要通过对系统的审计系统进行配置，就可使审计系统根据用户的要求监控、记录感兴趣程序的执行过程。执行迹中系统调用的次序关系是描述该程序行为的重要特征，而分析这种次序关系的最简单方法就是利用滑窗技术构造系统调用短序列。我们利用基于滑窗技术的程序执行迹预处理系统扫描每个监控进程的执行迹数据，为后继的进程行为分类器提供输入数据。

1 基于神经网络的行为分类器

人工神经网络具有自适应、自组织和自学习的能力，可以处理一些环境信息十分复杂、背景知识不清楚的问题。从模式识别的角度来看，入侵检测系统可以利用神经网络的学习能力来提取系统程序行为的模式特征，并以此为依据创建关于系统程序正常行为的特征轮廓[2]。由于系统中入侵行为的检测实际上也是对被监控系统及其实体(系统用户、系统进程等)的行为进行分类和识别的过程，因此可利用神经网络具有的自学习和并行处理能力，通过构造智能化的神经网络分类器来识别系统行为的异常性，从而达到检测系统中入侵行为的目的。

在设计基于神经网络分类器的检测系统时，不仅要考虑程序正常模式序列表中系统调用序列的唯一性，而且还要考虑到每个序列中系统调用的次序。

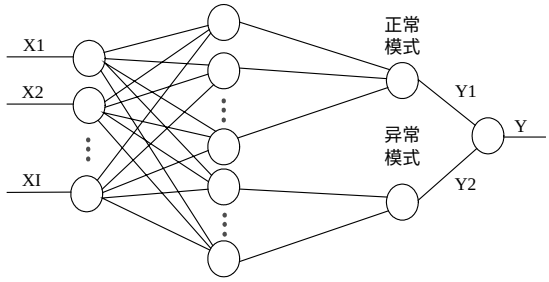

图 1 识别系统调用子序列正常与否的神经网络结构

作者简介：吴 玉(1965－)，男，硕士、高工，主研方向：计算机网络；陆晓君，硕士生、讲师
收稿日期：2006-02-23 **E-mail：**temothy2005@21cn.com

图 1 是本文所用的神经网络分类器的结构[3]。第一层为输入层，每个节点对应序列的一个元素，节点数与预处理系统输出的系统调用短序列的长度L一致。第二层为模式层(隐层)，模式层采用两种模式节点，分别代表进程执行迹的正常序列模式和异常序列模式。神经网络的模式层不仅描述了进程的正常执行特征轮廓，而且还包含了部分已知入侵行为的序列模式。这使得神经网络对进程行为的分类不仅是基于进程的正常执行特征轮廓，而且基于已知的入侵知识。不仅保证了对已知入侵行为的准确检出，而且对未知的入侵也具有相当的检测能力。这两种检测模式的结合，使网络能够学习正反两方面的知识，增大了分类器的检测准确度。模式层的每个节点代表一个模式，且与神经网络的所有输入层节点相连结。该层的主要功能是计算输入的序列向量$X=(x_1,x_2,...,x_l)$与模式层的某个模式(比如第j个模式$P_j=(p_{1j},p_{2j},...,p_{lj})$的匹配度，匹配函数($Match_j(X)$))定义如下：

$$Match_j(X)\begin{cases}1 & \sum_{i=1}^{l}(|x_i-p_{ij}|)=0\\ 0 & \sum_{i=1}^{l}(|x_i-p_{ij}|)\neq 0\end{cases}$$

若输入向量$X=(x_1,x_2,\ldots,x_l)$与模式层的某个模式匹配时，对应模式匹配函数的计算结果为 1，或者取值为 0。第三层为网络输出层，由分别标识为“正常”或“异常”的两个输出节点组成。模式层中代表程序正常模式的隐节点的计算结果则经过异常输出节点的累加汇总。然后这两个节点的输出结果经过进一步的汇总分析，作为分类器的输出。

根据上面定义，神经网络的模式层的每个节点都唯一地代表一个序列模式。所以，针对一个输入的系统调用短序列，最多只能有一个模式节点输出为 1，其他均为 0。也就是说，神经网络的正常和异常两个输出节点的输出不可能同时为 1，当正常节点输出为 1 时，说明输入序列为一正常序列。当异常节点输出为 1 时，说明被监控的程序已受到已知的入侵行为的攻击，可以立即报警或采用相应的措施。当两个输出节点都输出 0 时，说明输入的序列没有在已知的正常序列列表中，也不是已知入侵的特征模式，那么该序列可能是未知的正常序列模式(训练数据集的不完备时)或未知的入侵模式特征。这种不能明确判断的中间情况，需要对神经网络的检测结果做进一步的监控处理。而在具体实现时，为了简化网络输出层的处理，假设：不是正常的，就认为是异常的。

2 基于概率统计的贝叶斯分类器

贝叶斯分类算法通过计算一个待识别的序列属于哪个类的概率最大来确定它的归属[4]，具有一定的预测性，有利于系统对未知的系统正常行为的正确判断。基于贝叶斯定理的分类学习算法可以应用到入侵检测领域。贝叶斯分类学习算法是一种非常实用的分类算法，该算法不仅简单、实用，而且能够处理中、大规模的训练数据集。算法要求描述事件的属性之间具有条件独立性。实际情况下，虽然属性之间相互独立的条件不一定满足，但是实践证明，该算法仍然具有相当好的分类预测效果。因为在事例属于各类的后验概率都计算出来后，只是取其中概率值最大的那个类，所以在分类器工作时，无需非常精确地计算每一个事例分属哪一类的后验概率。这样，就可以有效地简化分类器中后验概率的计算问题。

贝叶斯分类算法描述如下：

假设目标函数：$f:X\rightarrow V$，其中X为事例集，V为函数的值域。应用到分类领域，则V就是事例的类别的集合。如果每个事例$x\in X$，由它的属性值组成的n元组$(a_1,a_2,...,a_n)$表示，那么事例x属于类别$v_j\in V$的概率为：$P(v_j|\ a_1,a_2,...,a_n)$。贝叶斯分类器定义目标函数f：

$$f(x)=v_{MAP}=\arg\max_{v_j\in V} P(v_j|\ a_1,a_2,\ldots,a_n)$$

$$=\arg\max_{v_j\in V}\frac{P(a_1,a_2,\ldots,a_n|v_j)P(v_j)}{P(a_1,a_2,\ldots,a_n)}$$

$$=\arg\max_{v_j\in V} P(a_1,a_2,\ldots,a_n|v_j)P(v_j)$$

为了简化计算，假设事例的属性是相互独立的，则有

$$P(a_1,a_2,\ldots,a_n|v_j)=\prod_{i=1}^{n}P(a_i|v_j)$$

由此可以得到简化的贝叶斯分类器：

$$v_{NB}=\arg\max_{vj\in V} P(vj)\prod_{i=1}^{n}P(ai|vj)$$

给定一个训练集合 X，贝叶斯分类器的学习算法为

NBCLA(X){

For(each $v_j\in V$){

采用统计方法，估算v_j在训练集合中出现的概率，记为：$\hat{p}(v_j)$；

针对事例集合中的每一个属性$Attr_i$的每一个取值$Attr_k$(属性$Attr_i$的第k个取值，$k\geq 1$)估算其在v_j类事例中出现的概率：$\hat{p}_i(Attr_k|v_j)$

}

}；

利用训练好的贝叶斯分类器对给定事例$x\in X$(由它的属性值组成的n元组$<a_1,a_2,\ldots,a_n>$表示)进行分类的算法：

Classify_New_Instance(x){

$$v_{NB}=\arg\max_{v_j\in V} P(v_j)\prod_{i=1}^{n}\hat{p}_i(a_i|v_j)$$

}

该算法给出了给定事例的分类。

3 基于进程行为分类的入侵检测系统原型

一般来说，在程序遭受到入侵攻击时，攻击时刻对应的系统调用附近的系统短序列大多会呈现异常。也就是说，表示入侵行为的异常性事件具有时间上的局部集中性，这为我们利用系统调用短序列判断进程是否受到入侵提供了依据。

由于分类器的不精确，对系统调用短序列的分类结果也会偶尔出现错误。为解决这个问题，可以通过分析某一时间段内，异常序列的百分比是否达到给定的门限，来判断系统进程执行迹是否异常。而在我们的程序识别器中，则采用反映最近事件状态的“漏桶”算法，利用入侵引起的异常事件在时间顺序上的局部集中性，对进程行为分类器的识别结果做进一步的分析，来减少误判的可能。

在实际系统中，不可能只监控一个系统关键程序，若同时监控多个系统程序，那么预处理程序可以通过进程标识符区别不同进程的执行迹(执行迹数据文件由二元组(进程标识符、系统调用序号)的序列组成)，但是并不能指出某个进程执行迹属于哪一个系统关键程序。所以预处理系统从系统审计迹中每识别出一个新的进程，就需要关于所有被监控程序的识别系统来对它进行检测，以确定它是否正常。因此我们

利用把多个识别不同程序的程序识别器，组合到一起的进程识别器，来判断一个进程是否是被监控程序集中某一个程序的正常执行，组成进程识别器的程序识别器个数与系统要监控的程序的个数相同。输入部分大家共用，对每个程序识别器的判断结果进行综合判断，只要有一个程序识别器认为系统审计的进程执行迹是正常的，那么检测系统就认为被监控的进程为正常执行的进程。只有所有的程序识别器都认为是异常的，那么检测系统就认为被监控的进程为正常执行的进程。只有所有的程序识别器都认为是异常时，才可以确定该进程是对某一个被监控程序的入侵攻击进程。

在图 2 中的z_i为进程检测器i的检测结果，即被检测的进程是否为某一个被监控程序的正常执行，得到的判断结果由入侵检测系统的报警及处理机制做进一步的处理：关闭对应的进程、给出报警信息等。

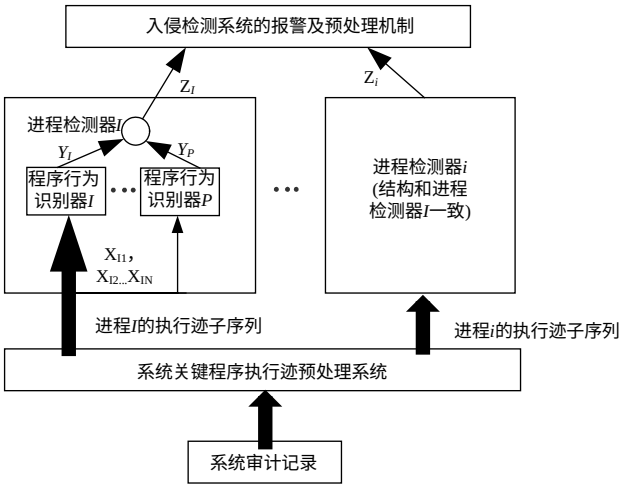

图 2 基于进程行为分类的入侵检测系统原型

设计要求系统审计系统只能对入侵检测系统中被监控的系统关键程序进行审计，否则由于每个程序都是可区分的，当系统审计系统对入侵检测系统没有提供监控的一个程序审计时，系统审计迹中就会包含该程序的执行情况的系统调用序列，入侵检测系统就会派生一个系统进程检测器对它进行处理，由于入侵检测系统中没有关于该程序正常执行特征轮廓的知识，因此就会把它的执行认为是一个受到入侵的进程而进行报警，甚至采取相应措施关掉该进程，造成用户的损失以及造成入侵检测系统过多的虚警现象。

4 结论

本文的神经网络分类器不仅基于进程的正常执行特征轮廓，而且基于已知的入侵知识。这使得它不仅保证了对已知入侵行为的准确检出，而且对未知的入侵也具有相当的检测能力。但因它采用模式匹配的方式，若训练数据不足，就会使得系统无法提取完备的正常行为模式集，造成较高的虚警率。而贝叶斯分类器，则利用概率统计以及机器学习理论对这些系统调用短序列进行处理。它通过计算一个待识别的序列属于哪个类的概率最大来确定它的归属，具有一定的预测性，有利于系统对未知的系统正常行为的正确判断。而且我们在设计入侵检测系统时，采用“漏桶算法”强调了异常系统调用短序列在时间上的局部相关性，忽略了分散、偶然的异常子序列的影响，从而有效地对分类的结果进行了综合分析，使得检测系统具有较高的检测性能。实践证明，基于该系统原型设计的入侵检测系统，能够有效地检测出那些改变系统程序行为的入侵攻击(例如特洛伊木马、缓冲区溢出以及病毒等)。

参考文献

1 张　琨. 用于入侵检测的贝叶斯网络[J]. 小型微型计算机系统, 2003, 24(5): 913-915.

2 橄书良. 基于神经网络的高效智能入侵检测系统[J]. 计算机工程, 2004, 30(10): 69-70.

3 杨立洁. 神经网络在入侵检测中的应用[J]. 济南大学学报(自然科学版), 2004, 18(1): 69-71.

4 李惠娟. 基于贝叶斯神经网络的垃圾邮件过滤方法[J]. 微电子学与计算机, 2005, 22(4): 107-110.

（上接第 148 页）

wav 声音信号。从图中可以看出嵌入前后音频信号没有太大的改变，实际上从听觉的角度也无法感知水印的嵌入。若直接从嵌入了水印的音频信号中进行提取，得到的结果如图 1 与图 2 所示。

为了验证算法的鲁棒性，我们对该嵌入水印的音频信号进行攻击处理。从经过被攻击的音频信号中提取出的水印图像可以看到，该图像有明显噪声，并且噪声成块状分布，原因是在对原始嵌入图像作预处理时采用的是分块的 DCT 变换。但是从它的归一化系数方面看还是在允许的范围以内的。提取水印结果如图 3 所示。

(a)　　(b)

图 3 滤波和加入高斯噪声后分别提取提取后的图像

从表 1 中可以看出，与传统的 DWT 方法相比，在抵抗以上两种攻击时，本文提出的方法表现了明显的优越性。

表 1 两种水印算法在相同条件下归一化相关值的比较

算法	无干扰	白噪声干扰	低通滤波	重采样
传统 DWT 算法	0.990 7	0.512 5	0.905 8	0.561 5
本文算法	0.992 5	0.909 7	0.950 4	0.566 9

参考文献

1 Kim W G, Lee J C, Lee W D. An Audio Watermarking Scheme with Hidden Signatures[C]//International Conference on Signal Processing, Beijing, China, 2000: 250-253.

2 钮心忻, 杨义先. 基于小波变换的数字水印隐藏与检测算法[J]. 计算机学报, 2000, 23(1): 21-27.

3 Swanson D, Zhu B, Tewfik H, et al. Robust Audio Watermarking Using Perceptual Masking[J]. Signal Processing, 1998, 66(6): 337-355.

4 Daubechies I. Ten Lectures on Wavelets[M]. Philadelphia: Society for Industrial and Applied Mathematics, 1992.

5 赵春晖, 李福昌. 一种DWT与DCT结合的盲音频水印算法[J]. 电子与信息学报, 2003, 25(11).