新的基于机器学习的入侵检测方法

田新广[1,3,4]，高立志[2,3]，张尔扬[1]

（1. 国防科技大学 电子科学与工程学院，湖南 长沙 410073； 2. 清华大学 电子工程系，北京 100084；
3. 北京首信集团 研究院，北京 100016； 4. 北京交通大学 计算技术研究所，北京 100029）

摘 要：提出了一种基于机器学习的用户行为异常检测方法，主要用于 UNIX 平台上以 shell 命令为审计数据的入侵检测系统。该方法在 Lane T 等人提出的检测方法的基础上，改进了对用户行为模式和行为轮廓的表示方式，在检测中以行为模式所对应的命令序列为单位进行相似度赋值；在对相似度流进行平滑时，引入了“可变窗长度”的概念，并联合采用多个判决门限对被监测用户的行为进行判决。实验表明，该方法在检测准确度和实时性上均优于 Lane T 等人提出的方法。

关键词：信息处理技术；入侵检测；机器学习；行为模式

中图分类号：TP18；TP393 **文献标识码**：A **文章编号**：1000-436X(2006)06-0108-07

Intrusion detection method based on machine learning

TIAN Xin-guang[1,3,4], GAO Li-zhi[2,3], ZHANG Er-yang[1]

(1. School of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China;
2. Department of Electronic Engineering, Tsinghua University, Beijing 100084, China;
3. Research Institute of IP Technology, Beijing Capitel Group Corporation, Beijing 100016, China;
4. Institute of Computing Technology, Beijing Jiaotong University, Beijing 100029, China)

Abstract: A new intrusion detection method was presented based on machine learning for intrusion detection systems using shell commands as audit data. In the method, multiple dictionaries of shell command sequences of different lengths were constructed to represent the normal behavior profile of a network user. During the detection stage, the similarities between the command sequences generated by the monitored user and the sequence dictionaries were calculated. These similarities were then smoothed with sliding windows, and the smoothed similarities were used to determine whether the monitored user’s behaviors were normal or anomalous. The results of the experience show the method can achieve higher detection accuracy and shorter detection time than the instance-based method presented by Lane T.

Key words: information processing technique；intrusion detection；machine learning；behavioral pattern

1 引言

网络入侵检测技术主要有 2 种基本类型：误用检测和异常检测。误用检测主要基于专家系统、模式识别等方法，目前已被广泛采用，其核心是根据已知的入侵和系统缺陷建立入侵模式库，通过被监测系统或用户的实际行为模式和入侵模式之间的比较或匹配来检测入侵，这种技术对已知入侵有很强的检测能力，但模式库需要不断更新，而且难以检测出未知入侵。异常检测则是对系统或用户的正

收稿日期：2004-05-11；**修回日期**：2006-04-24
基金项目：北京首信集团科研基金资助项目（011025）
Foundation Item: The Research Foundation of Beijing Capitel Group Corporation (011025)

常行为进行建模，当被监测系统或用户的实际行为同正常行为存在较大的差异时，即认为有入侵存在。异常检测的优点是不需要过多的有关系统缺陷的知识，具有较强的适应性，能够检测出未知入侵，但存在虚警概率高的缺点。

目前，异常检测是 IDS（入侵检测系统）的主要研究方向。国内外已经开展了机器学习、数据挖掘、神经网络等技术在异常检测中的应用研究[1~9]，研究目标主要是提高检测的准确性、实时性、高效性以及自适应性，其中一些研究成果已经接近或达到了实用化水平。美国普渡大学的 Lane T 等人以 UNIX 平台上的 shell 命令为审计数据，进行了基于机器学习的用户行为异常检测研究[1,2]；其研究内容是在用户界面层（user interface level）建立合法用户的正常行为轮廓，利用特定的相似度函数来计算该行为轮廓同用户当前行为模式的相似度，并将加窗平滑后的相似度作为检测异常行为的依据。中国科学院的连一峰等人提出了一种基于模式挖掘的用户行为异常检测方法[6]，该方法利用数据挖掘中的关联分析和序列挖掘技术对用户行为进行模式挖掘，采用基于递归式相关函数的模式比较算法对用户历史正常行为模式和当前行为模式进行比较，并根据比较结果来检测用户当前行为中的异常。

在 Lane T 等人的研究工作的基础上，本文提出一种新的基于机器学习的用户行为异常检测方法，该方法改进了对用户行为模式和行为轮廓的表示方式，采用了新的相似度计算（赋值）方法，并使用多个判决门限对用户行为进行判决，从而提高了用户行为模式和行为轮廓表示中的准确性和灵活性，增强了检测性能的稳定性和检测的实时性，并且降低了对主机系统资源的消耗。实验表明，该方法在检测准确度和实时性上均优于 Lane T 等人提出的检测方法。目前，该方法已经应用于国防科技大学和北京首信集团联合研制的主机型入侵检测系统（该系统已申请专利），并表现出良好的检测性能。

2　基本问题的描述与审计数据的处理

在一个实际的计算机网络系统中，一般会有多个合法用户。这些合法用户通常具有不同的操作权限（例如，程序员在系统中的主要活动是编程，而不允许执行网络管理员权限内的某些操作）。而且，不同的合法用户具有不同的行为特点和行为规律。在很多情况下，需要对系统中一些关键合法用户的行为进行监视，检测其行为中的异常，以防止其他用户（包括非法用户）冒用这些关键合法用户的账号进行非法操作，或者防止这些关键合法用户进行非授权操作。本文提出的用户行为异常检测方法在用户界面层建立网络系统中一个（或一组）关键合法用户的正常行为轮廓，并在检测中通过关键合法用户的当前行为与该正常行为轮廓之间的比较来识别异常行为；如果关键合法用户的当前行为较大程度地偏离了其历史上的正常行为轮廓，即认为发生了异常，这种异常可能是关键合法用户进行了非授权操作，也可能是系统中其他合法用户或外部入侵者（非法用户）冒充关键合法用户进行了非法操作。

本文提出的检测方法采用 UNIX 平台上的 shell 命令作为审计数据，其原因在于：1）在 UNIX 平台上，shell 是终端用户与操作系统之间最主要的界面，很大比例的用户活动都是利用 shell 完成的；2）同其他审计数据（如 CPU 使用量、内存占用率）相比，shell 命令能够更直接地反映用户的行为；3）shell 命令比较容易收集，也便于分析。

与 Lane T 等人的检测方法相同，本文的检测方法对用户在 shell 会话中所执行的原始 shell 命令行进行了预处理，具体操作如下：1）提取出 shell 命令的名称及参数，将 shell 命令行中的主机名、网址等信息用统一格式的标识符号来代替；2）将各命令符号按照在 shell 会话中的出现次序进行排列；3）把不同的 shell 会话按照时间顺序进行连接；4）在每个会话开始和结束的时间点上插入标识符号[1,2]。经预处理后，原始的 shell 命令行数据在形式上成为 shell 命令流。例如，某用户进行的 2 个时间上相邻的 shell 会话的命令行：

```
# Start session 1
cd ~/private/docs
ls -laF | more
cat foo.txt bar.txt zorch.txt > somewhere
exit
# End session 1
# Start session 2
vi scores.txt
mailx john_doe@somewhere.com
exit
# End session 2
```

经预处理后成为如下 shell 命令流：

（*SOF**，cd，<1>，ls，-laF，|，more，cat，

<3>，>，<1>，exit，**EOF**，**SOF**，vi，<1>，mailx，<1>，**EOF**）

其中**SOF**和**EOF**分别是一个会话开始和结束的标识符号，<1>、<3>为目录名（地址）符号。由此可见，shell 命令流是一系列按时间顺序排列的命令符号。

3 新的检测方法的描述及其特点分析

3.1 检测方法的描述

本文提出的用户行为异常检测方法利用多种长度不同的 shell 命令序列表示用户的行为模式，检测中以行为模式所对应的长度可变的 shell 命令序列为单位进行相似度计算（赋值）；在对相似度流进行平滑时，引入了“可变窗长度”的概念，并联合采用多个判决门限对被监测用户的行为进行判决。该方法的实现过程可以分为学习阶段和检测阶段。在学习阶段（训练阶段），需要定义多种长度不同的序列，针对每种序列建立一个样例序列库，用多个样例序列库来描述一个（或一组）合法用户的正常行为轮廓。建立样例序列库时，首先要得到该合法用户历史上正常操作时执行的 shell 命令行，并按照上述方法将 shell 命令行处理成 shell 命令流的形式（该命令流即为训练数据，它代表了该合法用户历史上的正常行为），然后，由该命令流生成多个命令序列流，并按照每个命令序列流中各序列的出现频率来提取相应的样例序列。学习阶段的工作具体分以下 3 个步骤进行。

step 1 定义 W 种长度不同的 shell 命令序列，用于表示一个（或一组）合法用户的各种行为模式。

设 W 种序列的长度的集合为 $C=\{l(1), l(2), \cdots l(W)\}$，其中 $l(i)$ 表示第 i 种序列的长度，且 $l(1)<l(2)<\cdots<l(W)$。在 W 确定的情况下，C 可有不同的选择。例如 $W=3$ 时，C 可以为 $\{1, 2, 3\}$（即 3 种序列的长度分别为 1, 2, 3），也可以为 $\{3, 6, 9\}$ 或其他组合。W 和 C 对检测性能有直接影响，在选择它们时，除了要充分考虑合法用户的行为特点之外，还需考虑检测系统的复杂度及检测效率（W 和 $l(i)$ 越大，检测系统的存储量和工作中的运算量也会越大）。

step 2 得到该合法用户历史上正常操作时执行的 shell 命令行，并将其处理成 shell 命令流的形式，然后由该 shell 命令流生成 W 个序列长度分别为 $l(1)$，$l(2)$，$\cdots$ $l(W)$ 的 shell 命令序列流。

设该 shell 命令流为 $R=(s_1, s_2, \cdots s_r)$，其中 s_j 表示按时间顺序排列的第 j 个 shell 命令符号，r 为该命令流的长度。这里，分别用 $S^1, S^2, \cdots S^W$ 表示由 R 生成的序列长度分别为 $l(1)$，$l(2)$，$\cdots$ $l(W)$ 的 W 个 shell 命令序列流，其中 S^i 是序列长度为 $l(i)$（$1\leqslant i\leqslant W$）的 shell 命令序列流；$S^i=(S_1^i, S_2^i, \cdots S_{r-l(i)+1}^i)$，其中 $S_j^i=(s_j, s_{j+1}, \cdots s_{j+l(i)-1})$。

step 3 从 W 个 shell 命令序列流中，分别按照序列的出现频率提取样例序列，建立 W 个样例序列库。

设 W 个样例序列库的集合 $L=\{L(1), L(2), \cdots L(W)\}$，其中 $L(i)$ 表示长度为 $l(i)$ 的序列对应的样例序列库。样例序列库集合 L 用于表示该合法用户的正常行为轮廓（正常行为模式的集合）。

定义 1 一个序列在序列流 S^i（$1\leqslant i\leqslant W$）中的出现频率等于该序列在 S^i 中的出现次数除以 S^i 中各个序列出现次数的总和。

提取样例序列并建立样例序列库的方法为：对于 W 个序列流 S^1，S^2，$\cdots S^W$，设定 W 个频率门限 η^1，η^2，$\cdots \eta^W$，其中 η^i（$1\leqslant i\leqslant W$）是序列流 S^i 对应的频率门限；将 S^i（$1\leqslant i\leqslant W$）中出现频率大于或等于 η^i 的命令序列视为合法用户的正常行为模式，并将这些命令序列提取出来作为样例序列，样例序列库 $L(i)$ 即是由这些命令序列组成（当有了新的训练数据时，可以根据新数据重新计算各序列的出现频率，进而对样例序列库作出调整。因此，这种方法对合法用户正常行为的变化是具有适应性的）。

在检测阶段(工作阶段)，首先要获取该用户在被监测的时间内执行的 shell 命令行，并按照上述方法将 shell 命令行处理成 shell 命令流的形式；然后，利用序列匹配方法在该 shell 命令流中进行“行为模式序列”挖掘，并对挖掘出的每个“行为模式序列”进行相似度赋值，从而得到一个相似度流；最后，对该相似度流进行加窗平滑处理，得到一系列的相似度判决值（相似度均值），并利用相似度判决值对该用户的当前行为进行判决。检测阶段的工作具体分以下几个步骤进行。

step 1 获取该用户（被监测用户）在被监测时间内执行的 shell 命令行，并将 shell 命令行处理成 shell 命令流的形式。设该 shell 命令流为 $\bar{R}=(\bar{s}_1, \bar{s}_2, \cdots \bar{s}_r)$，其中 $\bar{s}_j$ 表示按时间顺序排列的第 j 个

shell 命令符号，$\bar{r}$ 为该命令流的长度。在实时检测（在线检测）的情况下，$\overline{R}$ 中的每个 shell 命令符号是按照时间顺序依次得到的。

为分析方便，设 $\overline{R}$ 对应的序列长度为 $l(i)$ 的序列流为 $\overline{S}^i=(\overline{S}_1^i, \overline{S}_2^i, \cdots \overline{S}_{\bar{r}-l(i)+1}^i)$，其中 $\overline{Seq}_j^i=(\overline{s}_j, \overline{s}_{j+1}, \cdots \overline{s}_{j+l(i)-1})$。

step 2 利用序列匹配方法挖掘（定义）$\overline{R}$ 中的“行为模式序列”，并按照每个“行为模式序列”的长度对其进行相似度赋值。“行为模式序列”挖掘和相似度赋值的方法可描述如下：

步一：设定 $j:=1, i:=W$，$n:=1$。

步二：如果 $j \leqslant \bar{r}-l(W)+1$，将 $\overline{S}_j^i$ 同样例序列库 $L(i)$ 进行比较，然后执行步三；否则，停止操作（“行为模式序列”挖掘和相似度赋值的过程结束）。

步三：如果 $\overline{S}_j^i \in L(i)$（即 $\overline{S}_j^i$ 与 $L(i)$ 中的某个序列相同），则 $S_n^* := \overline{S}_j^i$，$\mathrm{sim}(S_n^*, L) := 2^{l(i)}/2^{l(W)}$，$j:=j+l(i)$，$i:=W$，$n:=n+1$，并返回执行步二；如果 $\overline{S}_j^i \notin L(i)$，则 $i:=i-1$，然后执行步四。（S_n^* 表示在 $\overline{R}$ 中挖掘到的第 n 个“行为模式序列”，$\mathrm{sim}(S_n^*, L)$ 表示 S_n^* 与样例序列库集合 L 的相似度）。

步四：如果 $i \neq 0$，返回执行步二；如果 $i=0$，则 $S_n^* := (s_j)$，$\mathrm{sim}(S_n^*, L) := 0$，$j:=j+1, i:=W$，$n:=n+1$，并返回执行步二。

按照以上方法进行“行为模式序列”挖掘和相似度赋值，可得到按时间顺序排列的“行为模式序列”流 $P=(S_1^*, S_2^*, \cdots S_M^*)$，以及相应的相似度流 $Z=(\mathrm{sim}(S_1^*, L), \mathrm{sim}(S_2^*, L), \cdots, \mathrm{sim}(S_M^*, L))$，其中 M 是从 $\overline{R}$ 中挖掘到的“行为模式序列”的个数，$\mathrm{int}(\bar{r}/\ l(W)) \leqslant M \leqslant \bar{r}-l(W)+1$（int 表示取整运算）。

step 3：对相似度流 $Z=(\mathrm{sim}(S_1^*, L), \mathrm{sim}(S_2^*, L), \cdots \mathrm{sim}(S_M^*, L))$ 进行加窗并取均值，将得到的相似度均值(判决值)与判决门限比较，进而对被监测用户的行为作出判决。在对相似度流进行加窗处理和对被监测用户行为进行判决时，有以下 2 种方案可以选择。

第一种方案：设定一个窗长度和一个判决门限，采用固定的窗长度对相似度流进行加窗并取均值，得到相似度判决值(相似度均值)，并利用相似度判决值和判决门限对被监测用户的行为作出判决。这种方案详细描述如下：

设窗长度为 w，判决门限为 λ。在对被监测用户的行为进行判决的过程中，当从 $\overline{R}$ 中挖掘到第 n 个“行为模式序列” S_n^* 并计算出 $\mathrm{sim}(S_n^*, L)$ 之后，就可以对相似度流中以 $\mathrm{sim}(S_n^*, L)$ 为终点的 w 个相似度（即 $\mathrm{sim}(S_{n-w+1}^*, L)$，$\mathrm{sim}(S_{n-w+2}^*, L)$，$\cdots$ $\mathrm{sim}(S_n^*, L)$）进行加窗并取均值，得到 S_n^* 对应的相似度判决值 $D(n)$

$$D(n)=\frac{1}{w}\sum_{m=n-w+1}^{n}\mathrm{sim}(S_m^*, L) \tag{1}$$

然后，利用判决值 $D(n)$ 和判决门限 λ 对被监测用户的当前行为进行判决。如果 $D(n)>\lambda$，将被监测用户的当前行为判为正常行为；如果 $D(n) \leqslant \lambda$，将被监测用户的当前行为判为异常行为。（这里，被监测用户的“当前行为”是相对于 S_n^* 而言的，它是指被监测用户执行的以 S_n^* 为终点的 w 个“行为模式序列”，即 S_{n-w+1}^*、S_{n-w+2}^*、$\ldots S_n^*$。）$D(n)$ 中，n 的初值为 w（即 $n \geqslant w$），n 的增长步长为 1（在挖掘到第 w 个“行为模式序列”之后，每挖掘到一个“行为模式序列”就可以对被监测用户的行为作一次判决）。当 $n<w$ 时，不计算 $D(n)$，也不作判决。

第二种方案：设定多个窗长度，针对每个窗长度设定两个判决门限（上限和下限）。采用可变的窗长度对相似度流进行加窗并取均值，得到相似度判决值(相似度均值)，然后通过相似度判决值与相应判决门限的比较来对被监测用户的行为作出判决。这种方案详细描述如下：

设定 V 个窗长度 $w(1)$、$w(2)$、$\cdots w(V)$，且 $w(1)<w(2)<\cdots<w(V)$；设定 V 个判决上限 $u(1)$、$u(2)$、$\cdots u(V)$ 和 V 个判决下限 $d(1)$、$d(2)$、$\cdots d(V)$，其中 $u(k)$ 和 $d(k)$ 是第 k 个窗长度 $w(k)$ 对应的判决上限和判决下限（$1 \leqslant k \leqslant V$），且 $u(1)>u(2)>\cdots>u(V-1)>u(V)=d(V)>d(V-1)>\cdots>d(2)>d(1)$。在对被监测用户的行为进行判决（分类）过程中，当从 $\overline{R}$ 中挖掘到第 n 个“行为模式序列” S_n^* 并计算出 $\mathrm{sim}(S_n^*, L)$ 之后，按照以下方法得到 S_n^* 对应的相似度判决值 $D(n)$，并对被监测用户的当前行为作出判决。

步一：设定 $k:=1$。

步二：将 n 同 $w(k)$ 进行比较。如果 $n < w(k)$，则不计算 $D(n)$，也不对被监测用户的当前行为进行判决，并不再执行下面的步骤。如果 $n \geqslant w(k)$，执行步三。

步三：计算 $N(n,k)$。

$$N(n,\ k) = \frac{1}{w(k)} \sum_{m=\ n-w(k)+1}^{n} \mathrm{sim}(S_m^*,\ L) \qquad (2)$$

$N(n,k)$ 是对相似度流 $Z = (\mathrm{sim}(S_1^*,\ L), \mathrm{sim}(S_2^*,\ L), \cdots \mathrm{sim}(S_M^*,\ L))$ 中以 $\mathrm{sim}(S_n^*,\ L)$ 为终点的 $w(k)$ 个相似度 $\mathrm{sim}(S_{n-w(k)+1}^*,\ L)$、$\mathrm{sim}(S_{n-w(k)+2}^*,\ L)$、$\cdots$ $\mathrm{sim}(S_n^*,\ L)$ 进行加窗并取均值后得到的相似度均值。

步四：判断是否满足判决条件：$N(n,k) > u(k)$。如果满足该条件，则 S_n^* 对应的相似度判决值定义为 $D(n) := N(n,k)$，并将被监测用户的当前行为判为正常行为（这里，被监测用户的当前行为是指以 S_n^* 为终点的 $w(k)$ 个“行为模式序列”，即 $S_{n-w(k)+1}^*$，$S_{n-w(k)+2}^*, \cdots S_n^*$）。至此，对用户当前行为的判决结束，不再执行下面的步骤。如果不满足 $N(n,k) > u(k)$，执行步五。

步五：判断是否满足判决条件：$N(n,\ k) \leqslant d(k)$。如果满足该条件，则 $D(n) := N(n,k)$，并将被监测用户的当前行为判为异常行为；至此，对用户当前行为的判决结束。如果不满足 $N(n,\ k) \leqslant d(k)$，执行步六。

步六：$k := k+1$，即 k 的值增加 1，并返回执行步二。

需要指出，在实时检测（在线检测）的情况下，被监测用户所执行的 shell 命令行（审计数据）的获取及预处理，“行为模式序列”的挖掘和相似度的赋值，对相似度的加窗处理，以及对用户行为的判决都是同步进行的。当被监测用户执行完一个“行为模式序列”后，就可以对该“行为模式序列”进行挖掘，并计算该“行为模式序列”对应的相似度，然后对该相似度和它前面的若干个相似度进行加窗处理（得到该“行为模式序列”对应的相似度判决值），进而对被监测用户的当前行为作出判决。

3.2 特点分析

同 Lane T 等人的检测方法相比，本文提出的检测方法有以下特点：

1) 利用多种长度不同的 shell 命令序列表示用户的行为模式，并建立多个样例序列库来描述用户的正常行为轮廓，这提高了用户行为模式和行为轮廓表示中的准确性和灵活性。而 Lane T 等人的检测方法用长度固定的命令序列表示用户的行为模式，由于实际中不同用户所具有的行为模式存在差异，同一用户完成不同行为模式时所执行的命令个数也不尽相同，所以，用长度固定的命令序列难以全面准确地表示出用户的行为模式；而且，Lane T 等人的方法在应用中不容易估算针对具体用户的最佳序列长度。

2) 采用了以 shell 命令序列为单位进行相似度赋值的方法，该方法主要关心被监测用户的当前命令序列所代表的行为模式是否能够同合法用户的某种行为模式完全匹配，不存在 Lane T 等人赋值方法中的模糊性，这不仅提高了检测准确度，而且增强了检测性能的稳定性。

3) 在相似度处理和用户行为判决的第二种方案中，采用了可变窗长度和多个判决门限，缩短了检测时间，提高了检测的实时性。

4 实验设计与结果分析

作者利用普渡大学公开发布的 shell 命令实验数据对以上方法的性能进行了实验，该数据包含 8 个 UNIX 用户在 2 年时间内的活动记录（实验数据的详细说明见文献[1]）。实验中采用了其中的 4 个用户 user1、user2、user3、user4 的数据，并且将 user2 设为合法用户，将 user1、user3、user4 设为非法用户。每个用户的 shell 命令流中各有 15 000 个命令，user2 的前 10 000 个命令作为训练数据用于建立样例序列库，而每个用户的后 5 000 个命令作为测试数据用于性能测试。实验的参数设置为 $W = 3$，$C = \{1, 2, 3\}$，$\eta^1 = \eta^2 = \eta^3 = 0.000\ 2$。在实验的测试阶段（检测阶段），采用上述的 2 种方案进行了相似度加窗和判决，第一种方案的窗长度 $w = 50$，判决门限 $\lambda = 0.5$；第二种方案设定 3 个窗长度，分别为 $w(1) = 40$，$w(2) = 50$，$w(3) = 60$；3 个判决上限分别为 $u(1) = 0.7$，$u(2) = 0.6$，$u(3) = 0.5$，3 个判决下限分别为 $d(1) = 0.3$，$d(2) = 0.4$，$d(3) = 0.5$。同时，作者还对文献[1, 2]中 Lane T 等人的检测方法进行了实验，实验中序列长度设为 3，窗长度设为

91，其样例序列库是由训练数据对应的 shell 命令序列流中出现频率大于或等于 0.000 2 的序列组成，并采用了文献[2]中所描述的相似度函数，归一化判决门限为 0.66。图 1 和图 2 分别给出了采用 Lane T 等人的检测方法和本文的检测方法（第二种方案）进行实验时所得到的相似度判决值曲线，图中上方的实线为合法用户 user2 对应的曲线，下方的虚线表示的是非法用户（user1、user3、user4）的曲线。可以看出，图 2 中合法用户的曲线同非法用户的曲线的可分性明显好于图 1。

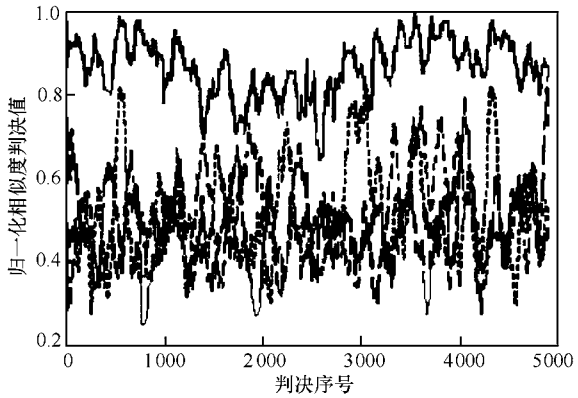

图 1　Lane T 等人的检测方法对应的判决值曲线

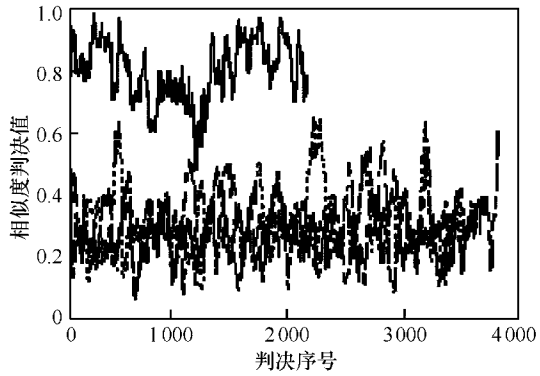

图 2　本文的检测方法对应的判决值曲线

在采用本文的方法进行实验时，样例序列库 $L(1)$、$L(2)$、$L(3)$分别由 120、350、670 个样例序列组成；测试阶段中从合法用户 user2 的测试数据（正常行为的测试数据）中挖掘出 1948 个“行为模式序列”，从 3 个非法用户 user1、user3、user4 的测试数据（异常行为的测试数据）中分别挖掘出 3 467，3 264，3 249 个“行为模式序列”。表 1 给出了以上实验的结果，表中虚警概率是指正常行为（user2 的行为）被错误判为异常行为的概率，检测概率是指异常行为（user1、user3、user4 的行为）被正确判为异常行为的概率，检测时间是指从被监测用户开始执行 shell 会话至检测系统对其行为作出判决的最短时间。

表 1　　**实验结果**

实验结果	虚警概率	平均检测概率	对异常行为的平均检测时间
Lane T 等人方法的实验结果	0.6%	91.7%	93.0 个 shell 命令符号的平均持续时间
本文方法中第一种方案的实验结果	1.0%	95.6%	75.2 个 shell 命令符号的平均持续时间
本文方法中第二种方案的实验结果	0.3%	96.1%	69.0 个 shell 命令符号的平均持续时间

根据表 1 的实验结果，本文的检测方法对异常行为的平均检测时间明显低于 Lane T 等人的方法，而且其检测准确度在总体上要好于 Lane T 等人的方法（只是第一种方案的虚警概率稍高一些）。此外，同第一种方案相比，第二种方案在检测时间和检测准确度上均具有较大的优势。

5　结束语

本文提出一种新的基于机器学习的用户行为异常检测方法，该方法具有较高的检测准确度和实时性，并且已经应用于实际的主机型入侵检测系统。需要指出，在实际应用中，通过优化参数设置和样例序列的表示及存储方式，还可以提高该方法的检测效率。

参考文献：

[1] LANE T. Machine Learning Techniques for the Computer Security Domain of Anomaly Detection[D]. Purdue University, 2000.

[2] LANE T, BRODLEY C E. An application of machine learning to anomaly detection[A].Proceedings of the 20th National Information Systems Security Conference[C]. 1997.366-377.

[3] LEE W, DONG X. Information-theoretic measures for anomaly detection[A]. Proceedings of the 2001 IEEE Symposium on Security and Privacy[C]. 2001. 130-134.

[4] WARRENDER C, FORREST S, PEARLMUTTER B. Detecting intrusions using system calls: alternative data models[A]. Proceedings the 1999 IEEE Symposium on Security and Privacy[C]. Berkely, California, USA：IEEE Computer Society, 1999. 133-145.

[5] KOSORESOW A P, HOFMEYR S A. A shape of self for UNIX

processes[J]. IEEE Software,1997,14(5)：35-42.

[6] 连一峰, 戴英侠, 王航. 基于模式挖掘的用户行为异常检测[J]. 计算机学报, 2002, 25(3):325-330.
LIAN Y F, DAI Y X , WANG H. Anomaly detection of user behaviors based on profile mining[J]. Chinese Journal of Computer, 2002, 25(3): 325-330.

[7] 孙宏伟, 田新广, 李学春. 一种改进的 IDS 异常检测模型[J]. 计算机学报, 2003, 26(11):1450-1455.
SUN H W, TIAN X G, LI X C. An improved anomaly detection model for IDS[J]. Chinese Journal of Computer, 2003, 26(11): 1450-1455.

[8] 田新广, 高立志, 李学春. 一种基于隐马尔可夫模型的 IDS 异常检测新方法[J]. 信号处理, 2003, 19(5):420-424.
TIAN X G, GAO L Z, LI X C. A new anomaly detection method based on hidden Markov models for IDS[J]. Signal Processing, 2003, 19(5): 420-424.

[9] 陈光英, 张千里, 李星. 基于SVM分类机的入侵检测系统[J]. 通信学报, 2002, 23(5):51-56.
CHEN G Y, ZHANG Q l, LI X. SVM classification-based intrusion detection system[J]. Journal of China Institute of Communications, 2002, 23(5):51-56.

作者简介：

田新广（1976-），男，河北吴桥人，北京交通大学博士后，主要研究方向为信号与信息处理、网络安全、入侵检测。

高立志（1966-），男，甘肃镇原人，清华大学电子工程系博士后，主要研究方向为数字信号处理、机器视觉与智能检测。

张尔扬（1941-），男，浙江宁波人，国防科技大学教授、博士生导师，中国通信学会会士，中国密码学会理事，主要研究方向为自适应信号处理、网络通信、卫星通信。

（上接第 107 页）

陈芳炯（1975-），男，广东汕头人，华南理工大学电信学院副教授，主要研究方向为信道盲估计、空时处理等。

韦岗（1963-），男，广西宾阳人，华南理工大学电信学院院长、教授、博士生导师，主要研究方向为信号处理和个人通信。