基于线性预测与马尔可夫模型的入侵检测技术研究

尹清波 张汝波 李雪耀 王慧强
(哈尔滨工程大学计算机科学与技术学院 哈尔滨 150001)

摘 要 入侵检测技术是现代计算机系统安全技术中的重要组成部分.该文提出了基于线性预测与马尔可夫模型相结合的入侵检测方法.首先提取特权进程的行为特征,引入时间序列分析技术——用线性预测技术对特权进程产生的系统调用序列提取特征向量来建立正常特征库,并在此基础上建立了马尔可夫模型.由马尔可夫模型产生的状态序列计算状态概率,根据状态序列概率来评价进程行为的异常情况.然后,利用马尔可夫信源熵与条件熵进行参数选取,对模型进行优化,进一步提高了检测率.实验表明该算法准确率高、实时性强、占用系统资源少.

关键词 线性预测;马尔可夫模型;入侵检测;马尔可夫信源熵;系统调用
中图法分类号 TP309

Research on Technology of Intrusion Detection Based on Linear Prediction and Markov Model

YIN Qing-Bo ZHANG Ru-Bo LI Xue-Yao WANG Hui-Qiang
(College of Computer Science and Technology, Harbin Engineering University, Harbin 150001)

Abstract Intrusion detection has emerged as an important approach of computer security technique. A new kind of method for anomaly intrusion detection is proposed based on linear prediction and Markov model. At first, linear prediction technique is employed to extract features from system call sequences of the privileged processes which are used to make up of the character database of those processes, and then the Markov model is founded based on the features; and Markov information source entropy and condition entropy are used to select parameter and optimize the model. The merits of the model are simple and exact to predict. The experiments show this method is effective and efficient in real time and light load, and can be used to in practice to monitor the computer system in real time.

Keywords linear prediction; Markov model; intrusion detection; Markov information source entropy; system call

1 引 言

入侵检测技术是现代计算机系统安全技术中的重要组成部分.它通过对计算机系统中的一些系统信息和系统中用户的一些行为信息进行分析来检测出对系统的入侵.

一个理想的入侵检测系统在具有 100 %攻击检

收稿日期:2003-08-24;修改稿收到日期:2005-02-01.本课题得到国家预研基金(413150702)和哈尔滨工程大学基础研究基金(HEUF04084)资助.尹清波,男,1975 年生,博士研究生,主要研究方向为信息系统安全理论与技术、机器学习、模式识别. E-mail: yinqingbo@hrbeu.edu.cn; yinq2004@126.com. 张汝波,男,1963 年生,教授,博士生导师,主要研究方向为信息系统安全理论与技术、机器学习、模式识别与人工智能. 李雪耀,男,1944 年生,教授,主要研究方向为模式识别与人工智能、语音信号处理. 王慧强,男,1962 年生,教授,博士生导师,主要研究方向为信息系统安全理论与技术.

测率的同时,误检率为零.并且要能在几乎不影响系统性能的前提下对计算机进行实时监测、实时保护.然而当前入侵检测系统存在共同的缺点:误检率(false positive rate)和漏检率(false negative rate)都很高.

入侵检测技术可以分为误用特征检测和异常检测两类.误用特征检测是根据搜集到的关于入侵或攻击的知识来检测入侵,即将当前行为与入侵模式库中的入侵模式特征相匹配来检测入侵.误用检测能够检测出绝大部分已知的入侵行为,但对于未知的入侵行为或已知入侵方法的变异却难以检测出来.并且把入侵活动用模式来描述也非常困难,而且很难检测出内部用户的攻击或误用;误用特征检测的性能和模式库的大小集体系结构有关.异常检测是基于对一个正常状态下系统的观察结果(系统、用户的统计模型)来检查当前状态对正常状态的偏离.尽管异常检测具有检测未知入侵的能力,但其误检率和漏检率往往都很高,使其迟迟不能被应用到实际当中来.

1996 年,Forrest 提出了一种通过监视特权进程的系统调用序列进行实时检测和分析的方法来对入侵行为进行检测[1].进程的行为可以用它发出的系统调用的序列来描述.对应于正常行为和异常行为的系统调用序列的统计特性不同,所以如果某进程发出的系统调用序列的统计特性和正常行为的统计特性有差别,则可以确定该进程有安全方面的威胁. Forrest 提出的方法是把正常的序列放在库中,将被监视进程的系统调用序列和正常库中的各个纪录进行匹配,如果匹配的比例比较大,则认为该进程进行的是正常的行为,否则认为是异常行为.

近几年,基于统计的学习方法得到发展,有基于频率统计、数据挖掘、有限自动机、神经网络、贝叶斯推理、支持向量机、隐马尔可夫模型、信息论方法等[2~16].这些方法除了基于隐马尔可夫模型比 Forrest 的模型好一些外,其它模型相差不多.这说明:(1)系统调用的规律性很强,简单的模型可以工作得很好;(2)模型的精确度还有待提高,即现有建模方法还不够精确.

Forrest 提出的方法,算法简单、容易实现,检测精确度高.然而,存在以下缺点:(1)正常行为没有和统计特性相结合;(2)建库及查询占用较多资源;(3)入侵者可以在其进程中插入一些无关紧要的系统调用来逃避检测;(4)没有为确定合理的短序列长度找到方法.

文献[7~9]为计算机系统的运行状态建立隐马尔可夫模型,但要么模型过于简单不能反映进程的内在规律(系统调用的逻辑关系不精确),要么算法过于复杂而检测结果并不理想.文献[10,11]为计算机系统的运行状态建立马尔可夫模型,但模型不是很精确,因此算法精度不高.

为了克服以上方法的缺点,本文提出了一种新的入侵检测方法——基于线性预测与马尔可夫模型的入侵检测技术(Linear Prediction and Markov Chain,LPMC).其基本思想是特定的程序有特定的、相对稳定的结构,因此将确定性方法(短序列建库)与随机性方法(马尔可夫模型)相结合来建立模型.首先从提取特权进程的行为特征入手,引入时间序列分析技术——用线性预测技术对特权进程产生的系统调用序列提取特征向量来建立正常特征库,并在此基础上建立了马尔可夫模型.由马尔可夫模型产生的状态序列计算状态概率,并根据状态序列概率来评价进程行为的异常情况.然后,利用马尔可夫信源熵与条件熵进行参数选取,并对模型进行优化,进一步提高了检测率.

2　LPMC 方法的原理

目前,绝大多数入侵行为都通过攻击特权进程来破坏计算机系统的安全性[1,11,12].直觉上,一定的系统调用排列应对应一定的程序功能,即程序行为的局部规律性应很强.特权进程通常完成特定的、有限的行为,所以其行为在时间上和空间上比其他用户程序要更稳定.并且入侵行为应具有某种功能行为特性,即系统调用序列应具有特定顺序排列.

对每一个系统调用赋予一个数值,则可以将系统调用序列看做是一个时间序列.因此,可以用数字信号处理与时间序列分析的方法来处理,然后对入侵和正常两种信号进行分类.这样,可以从短系统调用序列所要完成具体功能的确定性来提取局部特征;通过对进程的运行过程进行观察,利用随机过程和概率论的知识建立随机模型来描述系统调用序列(进程)的整体行为.

2.1　线性预测模型

线性预测(Linear Prediction,LP)的基本思想是:信号的每个取样值可以用它过去的若干个取样值的加权和(线性组合)来表示;各加权系数的确定原则是使预测误差的均方值最小.如果利用过去 q 个取样值来进行预测,称为 q 阶线性预测.

设 $x(n)$ 的预测值用 $\hat{x}(n)$ 表示,则有

$$\hat{x}(n) = -\sum_{l=1}^{q} a_l * x(n-l),\quad n \in N \tag{1}$$

式中,N 为正整数;$-a_l$ 表示加权系数,称为预测系数.

预测误差为 $e(n)$:

$$e(n) = x(n) - \hat{x}(n) = \sum_{l=0}^{q} a_l * x(n-l),\quad a_0 = 1,\ n \in N \tag{2}$$

预测误差按均方准则来确定:

$$\varepsilon_{\min} = E[e^2(n)] \tag{3}$$

对于特权进程的运行特征,可以对其建立时间序列模型,提取模型参数作为入侵检测特征,即用一组模型参数近似表达短系统调用序列的功能特征.关于 LP 模型的详细内容参见文献[17].

2.2 系统调用的马尔可夫模型

马尔可夫模型是一种简化的、高效的随机模型.马尔可夫模型假定在已知系统现在所处的状态的情况下,系统将来的演变与过去无关.特权进程的行为与其他用户进程相比是特定的、有限的、更稳定的,其运行过程中功能的转换与衔接也是有限的、比较稳定的.进程的正常操作和异常操作,其系统调用序列的概率分布是不同的,因此通过分析系统调用序列的统计性质就可以利用异常检测方法来判断该进程是否为入侵.计算机系统中的进程发出的系统调用序列的变化可以被近似地看作符合马尔可夫模型.计算机所发出的当前系统调用序列只与前一时刻发出的系统调用序列直接相关,而和前一时刻以前发出的系统调用序列不相关.因此可以用马尔可夫模型来描述进程的整体行为.

马尔可夫模型可以表示成一个三元组 $\Omega = (S, P, \pi)$.其中,S 是状态集合;$P:(S \times S) \to R_+$ 表示状态间的转移概率矩阵;π 表示系统的初始状态概率,且 $\pi = \{\pi_1, \pi_2, \cdots, \pi_s\}$.设随机变量序列 $\{X_n, n \in N\}$ 取值于状态空间 S,N 为正整数,有如下等式:

$$p(X_{n+1} = S_{n+1} \mid X_n = S_n, \cdots, X_0 = S_0) = p(X_{n+1} = S_{n+1} \mid X_n = S_n),\ S_0, S_1, \cdots, S_{n+1} \in S \tag{4}$$

若具有时齐性,则 $p(X_{n+1} = j \mid X_n = i) = p_{ij}, j, i \in S$ 且满足 $\sum_{j=1}^{s} p_{ij} = 1$.

可以将进程发出的系统调用序列的不同排列组合看作模型的不同状态,进程运行过程中功能的转换与衔接(对应不同组合的系统调用序列)可以作为状态间的转移.因此,进程的行为可以看作是马尔可夫模型的一系列状态之间的转换.则可以利用一系列状态的转换概率值来判别系统调用序列是否为异常.

3 LPMC 算法

用 Σ 表示所有系统调用的集合,用 O_{tr} 表示所有训练序列的集合,用 O_{te} 表示所有检测序列的集合.因为特权进程的行为与其他用户进程相比是特定的、有限的、稳定的,其运行过程中功能的转换与衔接也是有限的、稳定的,因此,其系统调用序列构成的集合是所有系统调用构成的序列的集合的真子集,记为 Σ^*,则 $O_{tr} \subseteq \Sigma^*$.对于任意序列 $\alpha \in \Sigma^*$,用 $|\alpha|$ 代表序列 α 的长度,且 α_i 表示 α 的前 i 个调用子序列,$\alpha[i]$ 表示 α 中第 i 个系统调用.

3.1 进程行为的特征提取与特征库的建立

设 w 为滑动窗口大小,用滑动窗口在序列 α 上滑动,步长为 1 个系统调用,则产生 $|\alpha| - w + 1$ 个短序列.对每一个短序列提取 q 阶 LP 系数 $(a_1, a_2, \cdots, a_q)$,$1 < q < w$.在本文中,用 Levinson-Durbin 算法进行快速迭带求取 LP 系数 $(a_1, a_2, \cdots, a_q)$[17].

将 $A_i = (a_1, a_2, \cdots, a_q)$ 作为与短序列相对应的特征向量.若 $\alpha \in O_{tr}$,将对应每个短序列的特征向量 $(a_1, a_2, \cdots, a_q)$ 作为纪录进行建库(剔除重复纪录),即若 A_i 为一个特征向量,则正常特征库为 $N_A = \{A_1, A_2, \cdots, A_n\}$,用 $|N_A|$ 表示特征库中的特征向量的数量.将特征库中的每个特征向量 A_i 作为马尔可夫模型的状态,即状态向量 S_i.

特征库 $N_A = \{A_1, A_2, \cdots, A_n\}$ 的规模由特征向量数决定,存储空间为 $O(q \times |N_A|)$.没有采用线性预测技术的特征库存储空间为 $O(w \times |N_A|)$.因为 $1 < q < w$,所以在提取了特征的同时也达到了压缩数据的目的,压缩比率由 q 与 w 决定.

3.2 进程行为的马尔可夫模型的训练

将特征向量 $(a_1, a_2, \cdots, a_q)$ 作为马尔可夫模型的状态,则马尔可夫模型状态空间为 $S \subseteq (E \times R^q)$,$E = \{e_i \mid e_i \in [-1, 1], 1 \le i \le q\}$.假设训练序列为 {a, s, d, f, e, g},且滑动窗口 $w = 4$,LP 阶数 $1 < q < w$,则状态与转移如图 1 所示.

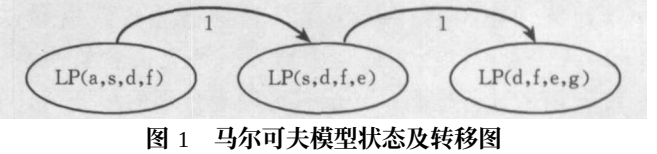

图 1 马尔可夫模型状态及转移图

马尔可夫模型状态转移矩阵 P 和初始分布 π 可由对以往状态的观察统计求得. 本文中用状态转移的频率来近似概率计算. 用计数器纪录每个状态出现次数 N_i ·、状态转换次序及次数 N_{ij}. 在训练序列中不可能包含正常序列的所有排列情况,即训练不可能是充分的. 这样有必要把状态分为两部分:必要状态、补充状态. 若 $\alpha \in O_{tr}$,则相应求得的 $(S_i = A_i) \in N_A$,称 S_i 为必要状态;若 $(\alpha \notin O_{tr}) \wedge (\alpha \in \Sigma^*)$,则有 $(\forall A_i \in N_A) \wedge (S_j \neq A_i)$,称 S_j 为补充状态. 为了正常库及状态规模扩展方便,设状态 S_0 表示补充状态. 则马尔可夫模型的一步状态转移概率矩阵为

(1) 必要状态一步状态转移概率:

$$p_{ij} = \frac{N_{ij}}{N_i \cdot},\ i \neq 0,\ j \neq 0 \text{ 且 } \sum_{j \neq 0} p_{ij} = 1.$$

(2) 补充状态一步状态转移概率:

$$p_{i0} = p_{0j} = \varepsilon \tag{5}$$

前面式中 N_{ij} 表示状态 i 向状态 j 转移的次数; N_i · 表示由状态 i 转移到任意一个状态的次数. 因为补充状态 S_0 在训练过程中没有出现过,因此有理由相信其出现的概率 ε 非常小,即要比出现过的状态的概率小很多,可使 $\varepsilon = \min(p_{ij})/10, 0 < i, j \leq |N_A|$.

经过大量实验观察发现,只要滑动窗口 w 设置得当,初始状态就为某一固定状态. 因此可设 $\pi = (1, 0, \cdots, 0)$.

3.3　检测方法

随着观测序列长度的不断增加,计算出的观测序列在正常情况下出现的概率会愈来愈小,很难据此概率的大小来判断观测序列是否正常,因此只有对相同长度观测序列进行比较才有意义. 检测算法的基本思想是建立在连续 L 个状态序列的转换概率上的. 即异常行为要实现入侵,其在行为的某一局部要实现特定的、与正常序列可区别的功能,依据状态序列概率值可以区别出正常、异常情况. 与建立马尔可夫模型的过程相同,得到要检测序列的连续 L 个状态,求其连续 L 个状态序列的转换概率.

$$P_t(S_{t-L+1}, \cdots, S_t) = \pi_{S_{t-L}} \prod_{i=t-L+1}^{L} p_{S_{i-1}S_i},\ 0 < L < t \tag{6}$$

存在两种异常情况:

(1) $S_{i-1} \rightarrow S_i$ 且 $(\forall A_i \in N_A) \wedge (S_i \neq A_i)$,即状态 S_i 不属于必要状态,则 $S_i = S_0$.

(2) $S_{i-1} \rightarrow S_i$ 且 $P(S_{i-1}, S_i) = 0$,则 $P(S_{i-1}, S_i) = \varepsilon$.

为了便于在线检测,可用如下递推公式:

$$P_t(S_{t-L+1}, \cdots, S_t) = \left(\frac{P_{t-1} * p_{S_{t-1}S_t}}{p_{S_{t-L}S_{t-L+1}}} \right),\ 0 < L < t \tag{7}$$

4　算法与性能分析

4.1　滑动窗口 w 的选取

4.1.1　对滑动窗口 w 的要求

对滑动窗口 w 的选取应满足两点:

(1) 确定性. 依据训练集求得的状态数,在增加训练序列后没有明显变化,即覆盖性.

(2) 随机性. 将状态作为变量,依据某种规则可以较好地描述与进程运行相对应的系统调用的随机性,即完整性描述.

4.1.2　马尔可夫信源熵准则

本文中利用马尔可夫模型来建立进程的随机模型. 同时可将系统调用序列看作是某一信号源发出的离散信号,且进程要完成特定的功能,则系统调用序列应有某种固定的排列结构. 由此可知,此离散信源应是有记忆的. 设信源所处的状态为 $S \in \{e_1, e_2, \cdots, e_j\}$,在每一状态下可能的输出符号为 $X \in \{x_1, x_2, \cdots, x_n\}$.

若信源满足以下两个条件,则称为马尔可夫信源.

(1) 某一时刻 t,信源符号的输出只与此刻信源所处的状态有关,与以前的状态和以前的输出符号无关. 即

$$P(X_t = x_k | S_t = e_i, X_{t-1} = x_{k_1}, S_{t-1} = e_j, \cdots) = p_t(x_k | e_i) \tag{8}$$

若具有时齐性,有 $p_t(x_k | e_i) = p(x_k | e_i)$.

(2) 信源在某 t 时刻所处的状态由当前的输出符号和前一时刻 $(t-1)$ 信源的状态唯一确定.

由条件知,若信源处于某一状态 e_i,当它发出一个符号后,所处的状态就改变了. 状态的转移依赖于发出的信源符号,因此任何时刻信源处于何状态完全由前一时刻的状态和发出的符号决定.

对 m 阶有记忆离散信源,它在任何时刻 t,符号发出的概率只与前面 m 个符号有关,把这 m 个符号看作信源在时刻 t 的状态. 信源输出依赖于长度为 $m+1$ 的随机序列就转化为对应的状态序列,而这种状态序列符合简单的马尔可夫模型的性质,可用马尔可夫模型来描述,称为 m 阶马尔可夫信源.

m 阶马尔可夫信源的极限熵 H_∞ 为

$$H_\infty = \lim_{N \to \infty} H(X_N | X_1 \cdots X_{N-1})$$

$$= \lim_{N\to\infty}\left\{ -\sum_{k_1}^{n}\cdots\sum_{k_N}^{n} p(x_{k_1}\cdots x_{k_N})\log_2 p(x_{k_N}|x_{k_1}\cdots x_{k_N})\right\}$$

$$= \lim_{N\to\infty}\left\{ -\sum_{k_1}^{n}\cdots\sum_{k_N}^{n} p(x_{k_1}\cdots x_{k_N})\log_2 p(x_{k_{m+1}}|x_{k_1}\cdots x_{k_m})\right\}$$

$$= \left\{ -\sum_{k_1}^{n}\cdots\sum_{k_{m+1}}^{n} p(x_{k_1}\cdots x_{k_{m+1}})\log_2 p(x_{k_{m+1}}|x_{k_1}\cdots x_{k_m})\right\}$$

$$= H(X_{m+1}|X_1 X_2\cdots X_m) \tag{9}$$

上式表明 m 阶马尔可夫信源的极限熵 H_∞ 就等于 m 阶条件熵,记为 H_{m+1}. 同时 $(x_{k_1}, x_{k_2}, \cdots, x_{k_m})$ 可表示为状态 e_i $(i = 1, 2, \cdots, n^m)$. 信源处于状态 e_i 时,再发下一符号 $x_{k_{m+1}}$,则信源从状态 e_i 转移到状态 e_j,即 $(x_{k_2} x_{k_3}\cdots x_{k_m} x_{k_{m+1}})$. 有

$$p(x_{k_{m+1}}|x_{k_1} x_{k_2}\cdots x_{k_m}) = p(x_{k_{m+1}}|e_i) = p(e_j|e_i) \tag{10}$$

则有

$$H_\infty = H_{m+1} = -\sum_i\sum_j p(e_i)\,p(e_j|e_i)\log_2 p(e_j|e_i) \tag{11}$$

其中 $p(e_i)$ 是 m 阶马尔可夫信源的状态极限概率. $p(e_j|e_i)$ 是状态一步转移概率.

由以上的讨论知,可依据 m 阶马尔可夫信源的熵的大小来确定滑动窗口 w,且 $w = m$. m 阶马尔可夫信源的熵值越小,说明模型的精确程度越高,同时建模过程中的时空开销也越大; m 阶马尔可夫信源的熵值越大,说明模型越粗糙,但建模过程中的时空开销也越小. 因此,模型的优化要在时空开销与模型精度间进行权衡,最终求得一个合理的折衷方案,来选择一个较合理的滑动窗口 w 的大小.

4.2 线性预测模型的阶数 q 的选取

线性预测模型拟合时间序列时,其准确性可以用 Akaike 的 FPE(Final Prediction Error)来衡量,其最小 FPE 对应的 LP 的阶数 q 就是最佳的模型阶数[18]. 但是通常可由一个关于滑动窗口 w 与线性预测阶数 q 的约束条件求得[19]:

$$0 < q \le 0.1w.$$

LP 的阶数 q 过大,则计算量过大. 从实时性考虑,在本算法中通常取 $q = 2$.

4.3 对马尔可夫模型状态长度 L 的选取

4.3.1 对马尔可夫模型状态长度 L 的要求

对马尔可夫模型状态长度 L 的选取应满足:

(1) 对正常进程序列求得的 L 个状态转移概率应与异常序列求得的 L 个状态转移概率有明显差别.

(2) 将马尔可夫模型连续 L 个状态可看作是时间序列,其规律性可由条件熵来衡量.

4.3.2 条件熵准则

定义条件熵为:令 $X = (e_1, e_2, \cdots, e_n)$, $Y = (e_1, e_2, \cdots, e_k)$,其中 $k < n$.

$$H(X/Y) = \sum_{x,y\in C_x C_y} P(x,y)\log\frac{1}{P(x/y)} \tag{12}$$

对于具有时间规律序列特征的数据,可以采用条件熵来衡量[13]. 在以前的研究中,用条件熵来衡量数据集(训练集、检测集)的规律性,这种规律性会影响检测器的性能;条件熵越小表示不确定性越小,根据这样的数据集合建立的模型的准确性越好. 在本文中,在训练数据集固定的条件下,用条件熵来求取马尔可夫模型的合理的状态序列长度 L.

令 $s_1, s_2, \cdots, s_n \in S$,则用下式计算条件熵 H_L:

$$H_L(s_L/s_1\cdots s_{L-1}) = -\sum_{s_L, s_1 s_2\cdots s_{L-1}} P(s_L, s_1 s_2\cdots s_{L-1})\log P(s_L/s_1 s_2\cdots s_{L-1}) \tag{13}$$

可以依据上式,对 L 的不同取值求取条件熵 H_L. 条件转移概率 $P(s_L/s_1 s_2\cdots s_{L-1})$ 及 L 阶先验概率 $P(s_1 s_2\cdots s_L)$ 可由以下两式求取条件概率与先验概率的近似值:

$$P(s_L/s_1 s_2\cdots s_{L-1}) = \frac{v(s_1 s_2\cdots s_{L-1} s_L)}{v(s_1 s_2\cdots s_{L-1})} \tag{14}$$

$$P(s_1 s_2\cdots s_L) = \frac{v(s_1 s_2\cdots s_L)}{v_0} \tag{15}$$

式中 $v(s_1 s_2\cdots s_{L-1} s_L)$ 是状态序列 $s_1 s_2\cdots s_{L-1} s_L$ 在训练集中出现的次数; v_0 是指训练序列的长度.

$$\Delta H = \left|\frac{H(s_L/s_1\cdots s_{L-1}) - H(s_{L+1}/s_1\cdots s_L)}{H(s_L/s_1\cdots s_{L-1})}\right| \tag{16}$$

根据信息论的知识,可以知道若马尔可夫模型的状态序列具有规律,则马尔可夫模型的状态序列长度 L 的值越大,条件熵 H_L 应该越小. 如果 ΔH 小于预先给定的域值 ε',即在增加状态序列长度 L 的情况下条件熵已经没有明显变化时,此时求得的状态序列长度 L 较合理.

4.4 性能评价相关的统计量

为了评价 LPMC 模型的检测性能,对相关的统计量作如下定义,并且为了与其它方法相比较将误检率分为两种情况进行定义:

检测率 TPR(True Positive Rate) = 检测出的异常序列数/异常序列总数;

漏检率 FNR(False Negative Rate) = (1 - 检测率);

误检率1- FPR(False Positive Rate) = 正常序列

被误报为异常序列数/正常序列总数;

误检率2-SFPR(Subsequence False Positive Rate) = 正常短序列被误报为异常短序列数/正常短序列总数.

5　实验及其结果

本实验采用美国新墨西哥大学计算机科学系提供的特权进程在正常情况下的系统调用序列及入侵过程中的系统调用序列①. 实验中,把 LPRCP 数据分为两部分:(1) 训练数据,共 3000 个正常进程的系统调用序列数据,用于建立正常进程的模型;(2) 测试数据,共有 1000 个正常进程,1000 个异常进程,这些测试数据用来测试模型与算法的有效性.

正常行为的特征库的规模(特征向量的个数),在训练数据固定的情况下随短序列长度 w(滑动窗口大小)的变化情况如图 2.

由图 2 可知,刚开始特征库的增长很快,但随后逐渐减慢. 当短序列长度 $6 < w < 12$ 时,且训练序列数大于 2500 时,特征库规模几乎不再增长. 由此可知,正常行为是有限的;且在训练数据有限的情况下,合理利用滑动窗口大小和训练集的关系,可以建立起一个较合理的模型来描述进程的行为.

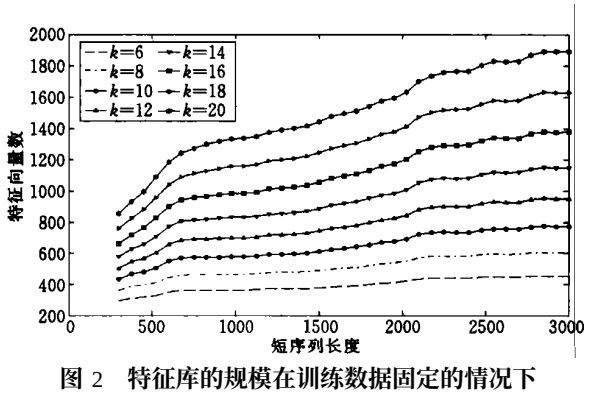

图 2　特征库的规模在训练数据固定的情况下随短序列长度 w 的变化情况

同时,进程由于环境等运行条件的不同,运行过程中产生的系统调用过程又具有随机性. 用马尔可夫模型建模后,利用马尔可夫信源的熵来衡量模型的好坏,进而决定出滑动窗口 w 的大小,如图 3 所示;利用条件熵来选择状态序列长度 L,如图 4 所示.

由图 3 知,滑动窗口 w 的最小值应为 5. 由图 4,可以发现对于状态长度 L 来说 4,8,10 是关键点,本文中取 $L = 10$.

为了进一步分析滑动窗口 w、马尔可夫信源的熵、建立的马尔可夫模型与检测性能之间的关系,将实验数据列为表 1.

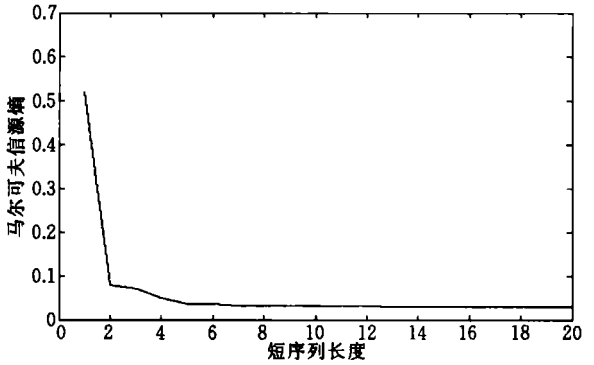

图 3　短序列长度与马尔可夫信源熵的关系

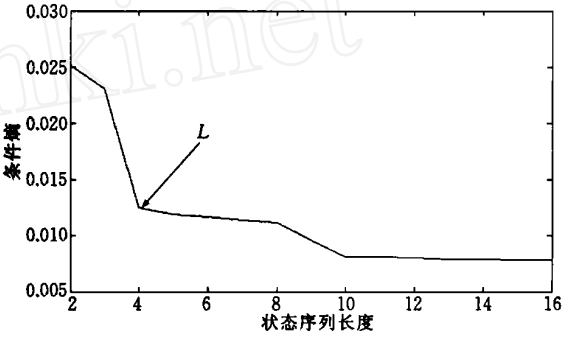

图 4　条件熵与状态序列长度 L 的关系

由表 1 知,当 $1 < w < 11$ 时,正常短序列与异常短序列出现的概率的差距逐渐拉大,马尔可夫信源的熵在明显减小,误检率明显减少;当 $11 < w < 20$ 时,正常短序列与异常短序列出现的概率的差距基本不变,马尔可夫信源的熵基本不变,误检率明显增大. 由以上分析可知,当马尔可夫信源的熵值明显减小时,检测性能提高;当马尔可夫信源的熵值基本保持不变时,检测性能变化不大或性能下降. 因此,可由马尔可夫信源的熵对建立的模型进行初步的参数选取与性能评价.

表 2 给出了 LPMC 的检测结果与 Mukkamala[5,6], Forrest[7], Yeung[15], 姚立红[16] 等人得出的研究结果的比较.

由表 2 中的数据,可以得出结论,本文的方法明显优于其它方法. 综合上述实验结果可知:

(1) 正常进程与异常进程的检测结果有明显差别;

(2) 随着滑动窗口的增大,正常进程与异常进程的差别也增大;

(3) 在训练数据有限的情况下,可利用马尔可夫信源的熵来限定滑动窗口的大小;

(4) 可利用马尔可夫信源的熵对模型的检测性能做初步的评价.

① http://www.cs.unm.edu/~immsec/

表 1 LPMC 的相关实验数据

w	L	连续 L 个正常训练短序列概率分布最小值	连续 L 个异常短序列概率分布最大值	TPR(%)	FPR(%)	SFPR	马尔可夫信源的熵
1	15	1.0764E-027	2.1583E-019	100	2.9	0.00040178	0.521860
2	10	1.7173E-016	6.7691E-012	100	0.4	4.5762E-005	0.078738
3	10	6.6088E-010	5.2520E-016	100	0.2	4.1280E-005	0.070583
4	10	1.5744E-008	2.6034E-023	100	0.2	3.2181E-005	0.050917
5	10	2.7014E-013	5.7548E-028	100	0.2	3.6863E-005	0.035787
6	10	2.2785E-008	2.6580E-034	100	0.2	3.2329E-005	0.034847
7	10	1.8004E-006	8.1691E-041	100	0.2	2.7775E-005	0.032011
8	10	1.7522E-006	1.7396E-047	100	0.2	2.3200E-005	0.031809
9	10	1.8553E-006	3.7931E-054	100	0.2	1.8603E-005	0.031367
10	10	1.5376E-006	7.9684E-061	100	0.2	1.3985E-005	0.031234
11	10	1.8670E-006	1.7603E-067	100	0.2	9.3450E-006	0.031114
12	10	1.7112E-006	1.7603E-067	100	0.3	1.6392E-005	0.030965
13	10	1.5795E-006	1.8453E-067	100	0.3	2.3472E-005	0.030509
14	10	1.2824E-006	1.8453E-067	100	0.4	3.2938E-005	0.030387
15	10	1.8524E-006	1.9294E-067	100	1.0	5.6599E-005	0.030373
16	10	1.8012E-006	1.9294E-067	100	1.2	8.5100E-005	0.030215
17	10	2.1308E-006	4.5789E-014	100	1.7	0.00042651	0.029414
18	10	2.0970E-006	2.0129E-067	100	1.3	0.00014488	0.029289
19	10	2.1055E-006	2.0960E-067	100	1.7	0.00018570	0.029201
20	10	2.1071E-006	2.0960E-067	100	1.7	0.00022671	0.029087

表 2 LPMC 与其它几种入侵检测方法的性能比较

方法	w	最高 TPR(%)	最低 FPR(%)	最低 SFPR
stide	6	96.9	—	0.0008
t-stide	6	96.9	—	0.0075
RIPPER	6	95.3	—	0.0016
HMM	6	96.9	—	0.0003
SVM	6	99.87	—	0.0003
CTBIDS	8	97.37	9/251	—
LPMC	6	100.00	2/1000	3.2329E-005
LPMC	11	100.00	2/1000	9.3450E-006

(5) 因为利用线性预测技术,可以使模型更精确且可以大量压缩数据来减小正常库的大小,提高了检测速度并节省了大量的系统资源;

(6) 该算法准确率高.

6 结 论

本文首先从特权进程的行为特征入手,引入时间序列分析技术——用线性预测技术提取特征向量来建立正常特征库,并在此基础上建立了马尔可夫模型. 为了精确描述进程运行的随机过程,提出了评价本模型、选取模型参数的准则与方法. 在此基础上又引入了马尔可夫信源的熵的概念,来确定滑动窗口 w 的大小. 最后将系统调用序列转变为状态序列,求取定长状态序列概率来判断是否异常,状态序列的长度 L 可由条件熵计算出. 马尔可夫信源熵的引入,对马尔可夫模型进行了优化,进一步提高了检测率.

实验结果表明,本方法对入侵进行检测非常有效. 今后,将进一步利用时间序列分析技术来获得更精确的模型.

参 考 文 献

1 Forrest S., Hofmeyr S. A., Somayaji A., Longstaff T. A.. A sense of self for unix processes. In: Proceedings of the 1996 IEEE Symposium on Security and Privacy. Oakland, California, 1996, 120～128

2 Lane T., Brodley C. E.. Temporal sequence learning and data reduction for anomaly detection. In: Proceedings of the 5th ACM Conference on Computer and Communications Security, San Francisco, California, 1998, 150～158

3 Lee Wenke, Stolfo S.J.. Data mining approaches for intrusion detection. In: Proceedings of the 7th USENIX Security Symposium, San Antonio, Texas, 1998, 79～93

4 Ilgun K., Kemmerer R., Porras P.. State transition analysis: A rule-based intrusion detection approach. IEEE Transactions on Software Engineering, 1995, 21(3): 181～199

5 Mukkamala S., Janowski G., Sung A. H.. Intrusion detection using neural networks and support vector machines. In: Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN), Honolulu, 2002, 1702～1707

6 Mukkamala S., Janowski G., Sung A. H.. Intrusion detection using support vector machines. In: Proceedings of the Advanced Simulation Technologies Conference, San Diego, 2002, 178～183

7 Warrender C., Forrest S., Pearlmutter B.. Detecting intrusion using system calls: Alternative data models. In: Proceed-

ings of the IEEE Symposium on Security and Privacy, Oakland, 1999, 133～145

8 Tan Xiao-Bin, Wang Wei-Ping, Xi Hong-Sheng, Yin Bao-Qun. A hidden Markov model used in intrusion detection. Journal of Computer Research and Development, 2003, 40(2): 245～250(in Chinese)
(谭小彬,王卫平,奚宏生,殷保群. 计算机系统入侵检测的隐马尔可夫模型. 计算机研究与发展, 2003, 40(2): 245～250)

9 Yin Qing-Bo, Zhang Ru-Bo, Li Xue-Yao, Wang Hui-Qiang. Research on technology of intrusion detection based on dynamic Markov model. Acta Electronica Sinica, 2004, 32(11): 1785～1788(in Chinese)
(尹清波,张汝波,李雪耀,王慧强. 基于动态马尔可夫模型的入侵检测技术研究. 电子学报, 2004, 32(11): 1785～1788)

10 Ye N.. A Markov chain model of temporal behavior for anomaly detection. In: Proceedings of the 2000 IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop, West Point, NY, 2000, 166～169

11 Jha S., Tan K., Maxion R. A., Roy A.. Markov chains, classifiers and intrusion detection. In: Proceedings of the 14th IEEE Computer Security Foundations Workshop, Cape Breton, Nova Scotia, 2001, 206～219

12 Hofmeyr S. A., Forrest S., Somayaji A.. Intrusion detection using sequences of system calls. Journal of Computer Security, 1998, 6(3): 151～180

13 Lee W., Dong X.. Information-Theoretic measures for anomaly detection. In: Proceedings of the 2001 IEEE Symposium on Security and Privacy, Oakland, California, 2001, 130～143

14 Eskin E., Lee W., Stolfo S.J.. Modeling system calls for intrusion detection with dynamic window sizes. In: Proceedings of the DARPA Information Survivability Conference and Exposition II (DISCEX II), Anaheim, CA, 2001, 165～175

15 Yeung D., Ding Yur-Xin. Host-based intrusion detection using dynamic and static behavioral models. Pattern Recognition, 2003, 36(1): 229～243

16 Yao Li-Hong, Zi Xiao-Chao, Huang Hao, Mao Bing, Xie Li. Research of system call based intrusion detection. Acta Electronica Sinica, 2003, 31(8): 1134～1137(in Chinese)
(姚立红,訾小超,黄　皓,茅　兵,谢　立. 基于系统调用特征的入侵检测研究. 电子学报, 2003, 31(8): 1134～1137)

17 Rabiner L., Juang B.. Fundamentals of Speech Recognition. Prentice-Hall International Inc, 1993

18 Akaike H.. A new look at statistical model identification. IEEE Transactions on Automatic Control, 1974, 19(6): 716～723

19 Thottan M., Ji C.. Adaptive thresholding for proactive network problem detection. In: Proceedings of the IEEE International Workshop on Systems Management, Newport, 1998, 108～116

YIN Qing-Bo, born in 1975, Ph. D. candidate, lecturer. His research interests currently focus on machine learning, pattern recognition, data mining, especially their applications in computer security like intrusion detection, user behavior modeling.

ZHANG Ru-Bo, born in 1963, Ph. D., professor and Ph. D. supervisor. His research interests include machine learning, pattern recognition, reinforcement learning and data mining.

LI Xue-Yao, born in 1944, professor. His research interests include pattern recognition, artificial intelligent and speech signal processing.

WANG Hui-Qiang, born in 1962, professor and Ph. D. supervisor. His research interests include information security theory.

Background

Intrusion detection is very important in the defense-in-depth network security framework and a hot topic in computer network security in recent years. An ideal intrusion detection system is the one that has 100% attack detection rate along with 0% false positive rate (the rate of mis-classified normal behavior), requires light load of monitoring, and involves minor calculation or overhead. Current intrusion detection systems, however, are plagued by either high false alarm probability or low attack detection accuracy.

This subject is supported by the project named "Resisting Intrusion and Reviviscence Technologies" and the project of the Harbin Engineering University Grants Committee named "Research on structural methods of knowledge discovery based on heuristic and its applications". This subject is motivated by current limitations of intrusion detection systems, which are generally unable to fully detect unknown attacks, or even unknown variations of known attacks, without generating a large number of false alarms. The focus of this project is to research new effective arithmetic and integrate intrusion detection with visualization techniques and human computer interaction strategies to address these limitations.