

文章编号:1001-9081(2006)05-1077-04

基于免疫原理与粗糙集理论的入侵检测方法

蒋世忠, 杨进, 张英
(广东药学院 计算机系, 广东 广州 510006)
(jiang6499@126.com)

摘要: 针对目前基于进程系统调用的入侵检测方法中存在的问题, 提出了一种基于免疫原理与粗糙集理论的入侵检测方法。该方法在对系统调用序列中的循环序列进行置换的基础上, 借助于粗糙集理论, 提取出一个简单的最小预测规则模型; 同时融合免疫原理的有关机制, 在检测模型中加入对已知入侵的快速检测引擎。同其他方法相比, 该方法不需要完备的进程系统调用数据, 而且得到的规则简单, 更适用于实时检测。实验结果表明, 该方法的检测效果优于同类的其他方法。

关键词: 入侵检测; 粗糙集理论; 免疫原理; 系统调用; 循环序列; 规则; 快速检测

中图分类号: TP309.2; TP393 文献标识码:A

Intrusion detection method based on immune and rough sets theory

JIANG Shi-zhong, YANG Jin, ZHANG Ying

(Department of Computer, Guangdong Pharmaceutical University, Guangzhou Guangdong 510006, China)

Abstract: Intrusion detection system has become the research hotspot because it can provide dynamic protection for computer system. Aiming at the problems existed in actual methods or models of intrusion detection, an effective method for intrusion detection based on immune theory and rough sets theory was presented in this paper. The circular sequences of system call sequences generated during the normal execution of a process is replaced by circular body, then, a little data is extracted from normal system call sequences, and is transformed to decisive table, afterward, the decisive table is reduced and the simplest rules that present normal behavior mode is extracted from reduct by rough sets theory. These rules can be used to detect anomalous behavior. In order to realize the quick detection of known intrusion, an engine of quick detection inspired by immune system theory was presented in this paper. Compared with other methods in the literature, the method presented in this paper is not only able to extract a set of effective detection rules with the minimum size from part of records of system call sequences, but also can detect the known intrusion quickly. Experiments show that this method in this paper is better than other methods based on system call.

Key words: intrusion detection; rough sets theory; immune theory; system call; circular sequence; rule; quick detection

0 引言

入侵检测系统因其能提供有效的动态保护而成为研究热点。根据数据分析方法和检测实现技术, 可以将入侵检测系统分为误用检测和异常检测。两种方法各有优缺点, 误用检测具有较低的虚警率, 但是只能发现已知的攻击, 而异常检测能够发现未知的攻击, 但是具有较高的虚警率。

1996 年, Forrest 等人借鉴免疫系统中 T 细胞形成的阴性选择机制提出了一种通过监视 UNIX/Linux 特权进程的异常检测新方法^[1]。Forrest 根据进程产生的系统调用短序列来建立进程的正常行为模式库, 然后通过它来监视进程的运行。Hofmeyr 在 Forrest 的基础上对正常库的结构进行了改进, 使效率得以提高^[2], Warrender 等比较了多种实现 Forrest 方法的效果^[3], W. Lee 等提出了用 RIPPER 算法^[4]来提取模式, 建立了一个简单有效的正常行为模式^[5], Wespi 等人提出了一种用可变长度的系统调用所产生的审计序列的短序列来刻画进程状态的异常检测模型^[6,7], 所得到的模式比固定长度短序列所产生的模式数目要少得多, Asaka 等提出了一种基于 Discriminant method 的检测方法, 通过对调用序列样本进

行学习来确定一个最优的分类面, 并利用此来判断异常^[8]。文献[9]将粗糙集理论应用于异常检测正常模型的建模过程, 从较小的正常系统调用中提取预测规则并用于异常检测。

研究人员对 Forrest 提出的方法进行了许多有益的探索, 并提出了一些部分解决问题的方法, 但这些方法大多需要有高质量的完备的系统调用样本数据进行训练, 在实际应用中, 能获得的训练样本数目不多, 而且绝大多数情况下只能拿到正常的训练样本。如果正常行为模式库不完备, 文献[1]和[7]中的方法在实用时就会产生大量的误报, 在实用中也有一定的困难。Lee 等提出的不能进行实时分析, 入侵的正确判断也依赖于系统调用序列的完备。文献[9]尽管是从小样本中提取规则, 但是没有考虑到系统调用中循环序列的影响, 样本系统调用数目仍较大, 所得到的规则也较多, 使得规则的提取和匹配时间延长。此外, 上面各方法均缺乏对已知入侵行为的准确及时的检测, 由于异常检测方法固有的特性, 上述各种方法均存在或高或低的虚警率, 其检测的准确性仍有待进一步地提高。

针对上述方法存在的问题, 本文在利用粗糙集进行规则提取前, 对系统调用中的循环序列进行置换处理, 进一步减少

收稿日期: 2005-11-10; 修订日期: 2006-01-10

作者简介: 蒋世忠(1972-), 男, 湖南邵阳人, 硕士, 主要研究方向: 人工智能、网络安全; 杨进(1978-), 男, 湖南邵阳人, 硕士, 主要研究方向: 网络安全、嵌入式系统; 张英(1974-), 男, 四川达州人, 硕士, 主要研究方向: 信息系统安全。

系统调用数目,使得规则的提取时间相应减少,得到的规则数也明显减少,从而提高系统的检测速度和准确度;借鉴免疫记忆与二次应答的思想,提出了入侵特征记忆库的概念,在异常检测中融合误用检测,实现对已知入侵行为的快速检测。实验结果表明,本文提出的方法不仅能够准确的区分正常行为与异常行为,而且能够快速地检测已知的入侵行为。

1 提取正常行为模式的粗糙集方法

1.1 建立正常系统调用决策表

文献[1]的实验表明,进程的正常行为可以用进程正常运行时的系统调用短序列来表征。设某进程在正常运行状态下产生的一条系统调用序列 W : write, sethostid, sstk, open, setuid, setgid, write;如果用长度为 4 的窗口在 W 上移动,可以得到该进程系统调用的短序列段集合 P : write, sethostid, sstk, open;sethostid, sstk, open, setuid;sstk, open, setuid, setgid;open, setuid, setgid, write;可以看出,对于每一个短序列段,只要知道短序列段前面三个系统调用,就可以准确预测出末位置的系统调用。

形式上,四元组 $S = (U, A, V, f)$ 是一个知识表达系统,也称为信息系统^[10],其中: U :对象的非空有限集合; A :属性的非空有限集合; $V = \bigcup_{a \in A} V_a$, V_a 是属性 a 的值域; $f: U \times A \rightarrow V$ 是一个信息函数,它为每个对象的每个属性赋予一个信息值,即 $\forall a \in A, x \in U, f(x, a) \in V_a$ 。

决策表定义如下:设 $S = (U, A, V, f)$ 为一知识表达系统, $A = C \cup D, C \cap D = \emptyset$, C 称为条件属性集, D 称为决策属性集。由决策表定义可知短序列段集合 P 实质上就构成一个可由条件属性决定决策属性的决策表,如表 1 所示。

表 1 正常系统调用组成的决策表

A1	A2	A3	D
write	sethostid	sstk	open
sethostid	sstk	open	setuid
sstk	open	setuid	setgid
open	setuid	setgid	write

1.2 提取系统调用的正常行为模式规则

在粗糙集理论中,设 $S = (U, A, V, f)$ 是一个决策表, U/C 表示 C 的所有等价类, U/D 表示 D 的所有等价类,令 X_i 和 Y_i 分别表示 U/C 与 U/D 中的各个等价类, $des(X_i)$ 表示对等价类 X_i 的描述,即等价类 X_i 对于各条件属性值的特定取值; $des(Y_i)$ 表示对等价类 Y_i 的描述,即等价类 Y_i 对于各决策属性值的特定取值。决策规则定义:

$$r_{ij}: des(X_i) \rightarrow des(Y_j), Y_j \cap X_i \neq \emptyset$$

参照决策规则的定义,在表 1 所示的正常系统调用的决策表中,可以提取出下面的 4 条规则:

- 规则 1: $A1(\text{write}), A2(\text{sethostid}), A3(\text{sstk}) \Rightarrow D(\text{open})$
- 规则 2: $A1(\text{sethostid}), A2(\text{sstk}), A3(\text{open}) \Rightarrow D(\text{setuid})$
- 规则 3: $A1(\text{sstk}), A2(\text{open}), A3(\text{setuid}) \Rightarrow D(\text{setgid})$
- 规则 4: $A1(\text{open}), A2(\text{setuid}), A3(\text{setgid}) \Rightarrow D(\text{write})$

上面 4 条规则就构成了正常行为模式规则集。那么,所得到的规则集是不是最简规则集呢?回答是否定的,对于决策表,还可以利用粗糙集理论中的约简方法进行简化,得到最简规则集。

1.3 提取行为模式的约简规则

众所周知,知识库中的知识并不是同等重要的,其中某些

知识甚至是冗余的。所谓知识的约简,就是在保持知识库分类能力不变的条件下,去掉其中不相关或不重要的知识。

令 $P \subseteq A$, 定义属性集 P 的不可区分关系 $ind(P)$ 为:

$$ind(P) = \{(x, y) \in U \times U | \forall a \in P, f(x, a) = f(y, a)\}$$

如果 $(x, y) \in ind(P)$, 则称 x 和 y 是 P 不可区分的, 令 R 为一族等价关系, $R' \in R$, 如果: $ind(R) = ind(R - \{R'\})$, 则称 R' 为 R 不必要的; 否则为必要的。如果每一个 $R' \in R$ 都是必要的, 则称 R 为独立的。设 $Q \in P$, 如果 Q 是独立的, 且 $ind(Q) = ind(P)$, 则称 Q 为 P 的一个约简。

利用粗糙集理论建立正常行为模式的方法就是建立系统调用的决策表,同时利用粗糙集理论的约简方法与规则生成方法,用简洁明了的规则来表达系统调用的正常行为模式。

1.4 循环序列的置换处理

在进程的系统调用序列中,某些短序列的反复、连续出现,表现出明显的循环性质,实验表明它影响所提取出的规则,从而影响检测的速度和准确度。在文献[7]的启发下,为消除循环所带来的影响,对其进行识别与置换。在本文中,将该类序列称为循环序列,将该短序列称为循环体。对于两个给定的循环序列,如果它们的循环体相同,就说明它们所执行的任务只有重复多少次的区别,没有任何实际内容上的差别。因此,每个循环序列也都可以用它的循环体来替换。

提取到循环体之后,以“循环体标识数”来标识循环体,对循环序列进行置换处理。所谓“循环体标识数”,就是用一个大于原映射表中的最大数字的数字来标识循环体,这个数字可以根据实际需要来选取,但必须保证它大于现有的系统调用总数,以避免重复。比如,对下面的系统调用序列: $X = 100, 109, 105, 104, 106, 105, 104, 106, 105, 104, 106, 56, 3, 2, 3, 2, 3, 2, 54, 32, 61, 32, 61, 32, 61, 85$ 。假设循环体的映射关系如下: 105, 104, 106 映射为正整数 400; 3, 2 映射为正整数 401; 32, 61 映射为正整数 402, 置换后得到下面的调用序列: $X' = 100, 109, 400, 56, 401, 54, 402, 85$ 。

由于循环体的长度不能太大,如果原序列中循环体中所含有的系统调用超过某个阈值后,循环序列对检测效果的影响已经微乎其微了,此时若再做置换就失去实际意义了。因此,可以选择一个比较合适的阈值 $LMAX$ 来限定循环体的长度。循环序列识别算法流程如下:

- 1) 查看以 s 为起始位置的序列中是否可能存在循环序列: 以循环体的长度 L (首次用 $LMAX$) 进行搜索;
- 2) 若以循环体长度 L 搜索未发现循环序列,则递增起始位置 s , 并转 1) 重新搜索;
- 3) 若搜索完序列仍未发现循环序列,则 $L = L - 1$, 若 L 大于等于 1, 转到 1); 否则退出算法;
- 4) 若对于起始点 s , 能够找到循环体长度为 L 的循环序列 S , 则保存 S 中包含的循环体及循环体的数目 k ;
- 5) 若循环体连续出现的次数 k 不小于限定值, 则确认已找到长度为 L 的循环体, 其出现次数为 k , 用循环体置换循环序列, 递增位置 s , 转到 4), 否则转到 3)。

2 基于免疫记忆和粗糙集的检测方法

2.1 入侵特征记忆库的建立

免疫系统的 B 细胞受抗原刺激后,部分 B 细胞成为对特异抗原具有高度亲和力的免疫记忆细胞。当免疫系统再次遭遇同种病原体时,由于免疫记忆细胞的存在,这些免疫细胞对

同种病原体具有高度的特异性和亲和力,无须重新学习,免疫系统能快速反击抗原,产生高滴度的抗体或效应细胞,清除进入体内的外部抗原,称为二次应答。受免疫系统中的免疫记忆、二次应答等机制的启发^[1],在实际中对系统进行攻击的入侵,多数是已知的入侵方法,为实现对已知入侵的快速检测,本文提出了入侵特征记忆库的概念。

从已知入侵的系统调用中提取出覆盖率较高的规则(相当于免疫记忆)加入入侵特征记忆库,实现当该入侵行为再次发生时,系统能迅速做出响应(相当于二次应答),加快对已知入侵的检测速度。建立入侵特征记忆库所使用的方法与正常行为模式库的建立方法类似,只不过不是从正常系统调用中提取规则,而是从已知入侵的系统调用中提取出其特有的规则。借鉴免疫记忆中免疫记忆细胞对曾经遇到过的病原体具有高度的特异性和亲和力的特点,只将覆盖率高的部分规则加入到入侵特征记忆库中,实现匹配的特异性,从而使得入侵特征记忆库中的规则数目大大减少,加快检测速度,达到免疫系统中二次应答机制所起到的同样效果。所谓规则的覆盖率,是指同该规则匹配的短序列段的数目占总序列段数目的百分比。

2.2 带快速检测引擎的检测算法

当某个待检测进程的系统调用序列经数据预处理进入到检测引擎时,首先使用检测引擎中的快速检测引擎对其进行入侵行为检测,并根据匹配数来决定是否为已知入侵行为,当快速检测引擎没有检测到已知入侵行为时,再将记录下来的调用短序列段与正常模式库中的规则进行匹配,并根据不匹配数来决定是否有异常行为发生。在检测时,相匹配的规则应该满足下面的条件:模式库中的规则与待检测短序列段的条件属性上的系统调用情况应该完全对应。下面给出检测算法:

第1步:用一个长度为 $K+1$ 的滑动窗口在待检测的系统调用序列上滑动,每一次滑动1步;如果是快速检测且没有检测到已知入侵,则进入第2步,如果快速检测已经检测到入侵行为,则算法结束,否则进入第6步;

第2步:在入侵特征记忆库的入侵行为模式中,寻找与待检测短序列段相匹配的规则;如果入侵特征记忆库中不存在与之相对应的规则,则用该进程在入侵特征记忆库中最常出现的结果作为前 K 个调用的预测结果;

第3步:如果在入侵特征记忆库中存在多条相匹配的规则,那么每一条相匹配的规则为其预测的结果投一票,累积票数最多者作为最后的结果;

第4步:如果入侵特征记忆库中规则预测的结果与待检测的短序列段的第 $K+1$ 个系统调用相同,则预测成功,匹配记数加1,若序列未结束,返回第1步;

第5步:如果入侵特征记忆库中的规则预测的结果与待检测的短序列段的第 $K+1$ 个系统调用不同,则此次预测失败,不匹配记数加1,若序列未结束,返回第1步;

第6步:在正常行为模式库中,寻找与待检测的短序列段相匹配的规则,如果正常行为模式库中不存在相对应的规则,则用最常出现的结果作为待检测短序列段中前 K 个系统调用的预测结果;

第7步:如果在正常行为模式库中存在多条相匹配的规则,那么每一条相匹配的规则为其预测的结果投一票,累积票数最多者作为最后的结果;

第8步:如果正常行为模式库中的规则预测的结果与待

检测的短序列段的第 $K+1$ 个系统调用相同,则预测成功,若序列未结束,返回第1步,否则算法结束;

第9步:如果正常行为模式库中的规则预测的结果与待检测的短序列段的第 $K+1$ 个系统调用不同,则此次预测失败,该进程的不匹配记数加1,若序列未结束,返回第1步,否则算法结束。

3 实验结果分析

3.1 实验说明

实验使用的数据,均来自 Forrest 在网站上提供的数据集^[12]。首先将原始序列中的循环序列进行置换,然后转化成给定长度的序列段,建立正常模式库时,从正常的数据集中随机抽取 2% 的序列段作为训练数据,剩余的 98% 的正常序列段和所有的异常序列段作为测试集;建立入侵特征记忆库时,从异常数据集中随机抽取 2% 的序列段作为训练数据,其余的数据作为测试集。实验数据经过预处理后填入训练决策表中,经属性约简后得到对应的约简表,然后基于约简生成预测规则集。最后,将训练得到的规则集应用于测试数据集,计算得到不同测试集的异常度。在每一窗口长度下,实验重复 10 次,平均后的实验结果经过归一化处理(将每种方法中异常度最大的值标定为 100)。初步实验表明,在窗口尺寸为 20 时,本方法得到最佳结果,由于篇幅原因,正常行为模式库只给出 sendmail 进程的比较结果,如表 2 所示。

表 2 正常模式库检测结果及比较

	本文	文献[9]	文献[1]	文献[5]
平均规则数目	692	843		
平均规则长度	7	7		
最大值	100	100	100	100
异常度	30.12	25.29	5.66	28.68
最小值				
“自我”误匹率	1.02	1.14	0	4.41

在提取出入侵行为的规则集后,计算出各规则的覆盖率,选取覆盖率高的规则加入入侵特征记忆库,并对入侵行为和正常行为进行检测,检测所得到的结果如表 3 所示。

表 3 入侵特征记忆库对入侵行为的快速检测结果

进程名称	数据集说明	窗口尺寸 20 时的异常度(%)	与正常序列表匹配度
sendmail	unsuccessful	42.23	3.32
	syslog	45.92	
xlock	vulnerability	44.26	2.84
	homegrow	39.01	
ps	recovered	38.53	1.63
	recovered	38.57	1.65
login	buffer overflow	31.04	1.18
named	UNM	55.76	4.07
lpr	MIT	52.29	

3.2 结果分析

(1) 从表 2 中的结果可以看到,非正常序列的异常度明显高于正常测试集的异常度。只要选取一个适当阈值,就能够准确地将进程的正常运行状态同异常运行状态区分开来。

(2) 表 2 给出了文献[1]、文献[5]和文献[9]中的检测结果。通过比较可以看到,本文提出的方法对异常行为的识别效果均要好于其他方法,对“自我”的识别要好于除文献[1]外的其他方法,至于文献[1]对“自我”的识别效果最好,是因为它将所有的正常序列用于训练,所以正常序列的异常度必然为 0,但是本文方法中的异常度最小值与正常序列匹配度的差值最大,因此本方法可以选取更小的阈值;与文献

[9] 比较,本文方法的规则数明显减少,检测时的比较次数减少,检测速度进一步提高。基于以上比较,本文提出的模型能更有效地将正常和异常序列区分开来方法,在给定的误报水平上,不仅可以降低阈值,进一步减少漏报,而且检测速度更快,适应在线检测。

(3) 由表 3 中的数据可以看出,尽管模型只采用了入侵行为中的部分规则,仍能够有效地区分入侵行为与正常行为,但是检测时规则的比较次数显著减少,达到快速检测的目的。

4 结语

在本文中,利用粗糙集来建立进程的正常行为模式库,只采用了进程所产生的部分正常数据来训练,就能得到一个简单的正常预测规则模型,同时借鉴免疫原理中的二次应答和免疫记忆机制,使用部分规则来建立入侵行为特征记忆库,最后通过实验证明了本文提出的模型是切实可行的。

参考文献:

- [1] FORREST S, HOFMEYR SA, SOMAYAJI A, et al. A Sense of self for unix processes[A]. Proceedings of the IEEE Symposium on Security and Privacy[C]. Oakland: IEEE, 1996. 120 – 128.
- [2] HOFMEYR SA, FORREST S, SOMAYAJI A. Intrusion detection using sequences of system calls[J]. Journal of Computer Security, 1998, (6): 151 – 180.
- [3] WARRENDER C, FORREST S, PEARLMUTTER B. Detecting in-

trusions using system calls: alternative data models[A]. IEEE Symposium on Security and Privacy[C]. CA: IEEE Computer Society, 1999. 133 – 145.

- [4] COHEN WW. Fast effective rule induction[A]. Proceedings of the 12th International Conference[C]. California, 1995. 115 – 123.
- [5] LEE W, STOLFO S, CHAN P. Learning patterns from unix Process execution traces for intrusion detection[A]. Proceedings of AAAI Workshop on AI[C]. Menlo Park, CA: AAAI, 1997. 50 – 56.
- [6] DEBAR H, DACIER M, WESPI A. Towards a taxonomy of intrusion detectionsy stems[J]. Computer Networks, 1999, 31(8): 805 – 822.
- [7] WESPI A, DACIER M, DEBAR H, et al. Intrusion detection using variable-length audit trail patterns[A]. Proceedings of the 3rd International Workshop on the Recent Advances in Intrusion Detection (RAID' 2000)[C]. Toulouse: 2000. 110 – 129.
- [8] ASAKA M, ONABUTA T, INOUE T, et al. A new intrusion detection method based on discriminant analysis[J]. IEICE Transactions on Information and Systems, 2001, E84-D(5): 570 – 577.
- [9] 蔡忠闽,管晓宏,邵萍,等. 基于粗糙集理论的入侵检测新方法[J]. 计算机学报,2003,(3): 361 – 366.
- [10] 张文修,吴伟志,梁吉业,等. 粗糙集理论与方法[M]. 北京: 科学出版社,2001.
- [11] 白惠卿,陈育民. 医学免疫学和微生物学[M]. 北京: 北京医科大学出版社,2003.
- [12] <http://www.cs.unm.edu/~immsec/data-sets.html>[DB/OL].

(上接第 1076 页)

训练,从余下的数据中随机选择 100 000 条记录作为测试集,用于测试系统的性能。

RBFIDS 系统采用检测率(DR)和误报率(FP)来评价系统的性能。其中:

$$DR = \frac{\text{正确检测的入侵数量}}{\text{入侵的总数量}} \quad (4)$$

$$FP = \frac{\text{被误报的正常数据}}{\text{总的正常数据}} \quad (5)$$

ROC(Receiver Operating Characteristic) 曲线是一种在入侵检测研究领域常用的结果分析工具。它以误分率(false positive rate, 正常数据错误分类到异常类的比例)作为 X 轴,以检测率(detection rate, 检测到的入侵事件占总体的比例)作为 Y 轴构成,可以反映入侵检测系统检测率和误报率之间的变化关系。针对以上的数据集,RBFIDS 系统的 ROC 曲线如图 3 所示。

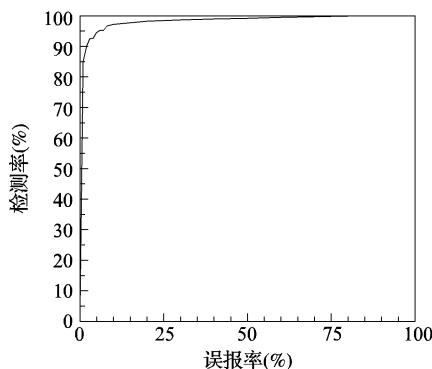


图 3 RBFIDS 系统的 ROC 曲线

3 结语

本文将径向基函数神经网络引入入侵检测系统,构造了

一个基于径向基函数的入侵检测系统,并论述了 RBFIDS 系统的系统结构和实现方法。使用 KDD99 数据集对 RBFIDS 系统进行性能测试,系统总检测率达到 98%,误报率仅为 1.6% (见图 3),表明将径向基函数神经网络引入到入侵检测中是成功的,系统具有较高的检测率和较低的误报率。

参考文献:

- [1] BIVENS A, PALAGIRI C, SMITH R, et al. Network – based intrusion detection using neural networks[A]. Proceedings of the Artificial Neural Networks in Engineering Conference: Smart Engineering System Design[C], 2002.
- [2] BOLANOS RF, CADENA CA, NINO F. Detection of denial of service attacks using neural networks[A]. 6th World Multiconference on Systemics, Cybernetics and Informatics[C], 2002.
- [3] LI J, MANIKOPOULOS C. Early statistical anomaly intrusion detection of DOS attacks using MIB traffic parameters[A]. IEEE Systems, Man and Cybernetics Society Information Assurance Workshop [C], 2003.
- [4] LI J, ZHANG GY, GU GC. The research and implementation of intelligent intrusion detection system based on artificial neural network [A]. Proceedings of 2004 International Conference on Machine Learning and Cybernetics[C], 2004.
- [5] ELFADIL N, ISA D. Automated knowledge acquisition based on unsupervised neural network and expert system paradigms[M]. Oxford, United Kingdom: Springer Verlag, Heidelberg, D-69121, Germany, 2003.
- [6] SING JK, BASU DK, NASIPURI M, et al. Improved k-means algorithm in the design of RBF neural networks[M]. Bangalore, India: Institute of Electrical and Electronics Engineers Inc, 2003.
- [7] STOLFO SJ, WEI F, WENKEL, et al. Cost-based modeling for fraud and intrusion detection: results from the JAM project[M]. Hilton Head, SC, USA: IEEE Comput. Soc, 1999.