计算机研究与发展 ISSN 1000-1239/CN 11-1777/TP
Journal of Computer Research and Development 47(3): 485－492, 2010

在线自适应网络异常检测系统模型与算法

魏小涛[1] 黄厚宽[2] 田盛丰[2]
[1](北京交通大学软件学院 北京 100044)
[2](北京交通大学计算机与信息技术学院 北京 100044)
(weixt@bjtu.edu.cn)

An Online Adaptive Network Anomaly Detection System-Model and Algorithm

Wei Xiaotao[1], Huang Houkuan[2], and Tian Shengfeng[2]
[1](*School of Software, Beijing Jiaotong University, Beijing* 100044)
[2](*School of Computer and Information Technology, Beijing Jiaotong University, Beijing* 100044)

Abstract The extensive usage of Internet and computer networks makes security a critical issue. There is an urgent need for network intrusion detection systems which can actively defend networks against the growing security threats. In this paper, a light weighted online adaptive network anomaly detection system model is presented. The related influence function based anomaly detection algorithm is also provided. The system can process network traffic data stream in real-time, gradually build up its local normal pattern base and intrusion pattern base under a little supervising of the administrator, and dynamically update the contents of the knowledge base according to the changing of the network application patterns. At the checking mode, the system can detect not only the learned intrusion patterns but also the unseen intrusion patterns. The model has a relatively simple architecture, which makes it efficient for processing online network traffic data. Also the detecting algorithm takes little computational time and memory space. The system is tested on the DARPA KDD 99 intrusion detection datasets. It scans 10% of the training dataset and the testing dataset only once. Within 40 seconds the system can finish the whole learning and checking tasks. The experimental results show that the presented model achieves a detection rate of 91.32% and a false positive rate of only 0.43%. It is also capable of detecting new type of intrusions.

Key words network anomaly detection; online adaptive; influence function; data stream; anomaly detection

摘 要 随着因特网等计算机网络应用的增加,安全问题越来越突出,对具有主动防御特征的入侵检测系统的需求日趋紧迫.提出一个轻量级的在线自适应网络异常检测系统模型,给出了相关算法.系统能够对实时网络数据流进行在线学习和检测,在少量指导下逐渐构建网络的正常模式库和入侵模式库,并根据网络使用特点动态进行更新.在检测阶段,系统能够对异常数据进行报警,并识别未曾见过的新入侵.系统结构简单,计算的时间复杂度和空间复杂度都很低,满足在线处理网络数据的要求.在DARPA KDD 99入侵检测数据集上进行测试,10%训练集数据和测试集数据以数据流方式顺序一次输入系统,在40 s之内系统完成所有学习和检测任务,并达到检测率91.32%和误报率0.43%的结果.实验结果表明系统实用性强,检测效果令人满意,而且在识别新入侵上有良好的表现.

收稿日期:2009-03-04;修回日期:2009-10-22
基金项目:国家自然科学基金项目(60442002)

关键词 网络入侵检测;在线自适应;影响度函数;数据流;异常检测

中图法分类号 TP393.08

计算机入侵检测技术分为误用检测和异常检测2种.误用检测是通过监视目标系统的特定行为与已知的入侵模式是否匹配来进行入侵检测的;而异常检测则是事先建立被监视目标在正常情况下的行为模式,通过检测当前行为是否显著偏离了相应的正常模式来进行入侵检测的.异常检测由于不需要入侵的先验知识并能够捕获以前未知的新入侵而受到广泛的重视.另外,从处理数据的来源看,入侵检测又分为基于网络的入侵检测和基于主机的入侵检测.本文研究的是基于网络的异常检测.

对于网络异常检测系统而言,除了要有较高的检测率外,从实用性的角度看还应满足:

1)系统结构简单、效率高,检测算法计算量小,适于处理在线网络数据;

2)具有自学习自适应能力;

3)具有较强的检测新入侵的能力;

4)具有较低的误报率,大量的误报会使系统的可用性降低.

针对上述要求本文提出一个在线自适应网络异常检测系统.系统能够处理实时网络数据流,其学习和检测是一个统一的过程,而且无论学习阶段还是检测阶段都只扫描数据一次;自适应是指系统能够动态构建和维护自身的知识库,能随着网络自身应用特征的改变而更新知识.在KDD99数据集上的实验结果验证了系统的效果和性能.

1 相关工作

网络异常检测方法的研究从1990年Heberlein等人开发的NSM(network security monitor)[1]系统开始.迄今为止,主要有概率统计分析方法、数据挖掘方法和生物系统模拟(神经网络、遗传规划、人工免疫系统等)方法等.

统计分析技术在入侵检测系统中的应用研究主要集中在马尔可夫模型和支持向量机模型上.Callegari[2]等人在网络异常检测中比较了几种不同的随机模型,包括一阶和高阶齐次马尔可夫链、非齐次马尔可夫链、稳定性和非稳定性经验累积分布函数等,实验结果显示高阶齐次马尔可夫链是效果最好的,但是模型的参数较难确定;文献[3]利用一阶齐次马尔可夫链对主机系统中特权程序的正常行为进行建模,并基于状态序列的出现概率判断异常行为;文献[4]使用了一个变长马尔可夫模型来捕获入侵轨迹的特征,对入侵行为进行实时预测.在使用支持向量机进行异常检测中,为了提高支持向量机的训练速度,文献[5]提出了一种增量学习算法,而文献[6]采用了对训练数据预先聚类的方法,都取得较好检测率,但是误报率很高.统计分析方法基本上不需要太多关于入侵技术细节的先验知识,而且有较为成熟的统计技术可以应用;但是漏报率和误报率都还较高,大部分的方法仍然需要干净的训练数据,这在真实的网络环境中很难确保.

基于数据挖掘的检测技术使用关联规则、序列挖掘、数据分类和聚类等算法从大量的网络数据中自动生成简洁而精确的检测模型.文献[7]使用频繁项集挖掘算法和衰减窗口技术来发现网络数据流的应用模式,能够高效学习,缺点是不能检测新入侵,检测率较低;文献[8]首先将训练样本进行聚类,然后在每一聚类上训练一棵ID3决策树;文献[9]针对每一类入侵训练一棵两类决策树,检测时将分类结果进行组合,并通过提升技术改进其检测性能.这些方法都取得了较好的结果,但是基于数据挖掘的方法往往需要大量的有标号数据作为基础,系统比较复杂,在检测模型学习和评价阶段的计算成本高,难以实现系统的实时学习.

基于生物系统模拟的方法最近集中在分布式神经网络和分布式遗传规划上.文献[10]将大数据集随机分割成小块并使用分布式神经网络进行并行学习,用于大规模网络入侵检测,取得较高检测率,但误报率较高;文献[11]使用分布式遗传规划方法训练决策树分类器,并通过提升的方法分配各分类器的权重,有效降低了误报率.但这些算法在学习过程中同样需要大量带标号数据,且计算复杂度较大.

总之,异常检测技术仍然面临检测率低和误报率过高的问题,并且多数模型系统结构复杂、效率低,难以适应在线检测的要求.

2 系统模型与算法

在线自适应网络异常检测系统模型如图1所示,系统分为4个部分:(A)数据预处理模块;(B)模式匹配与更新模块;(C)决策模块;(D)报警与响应模块.

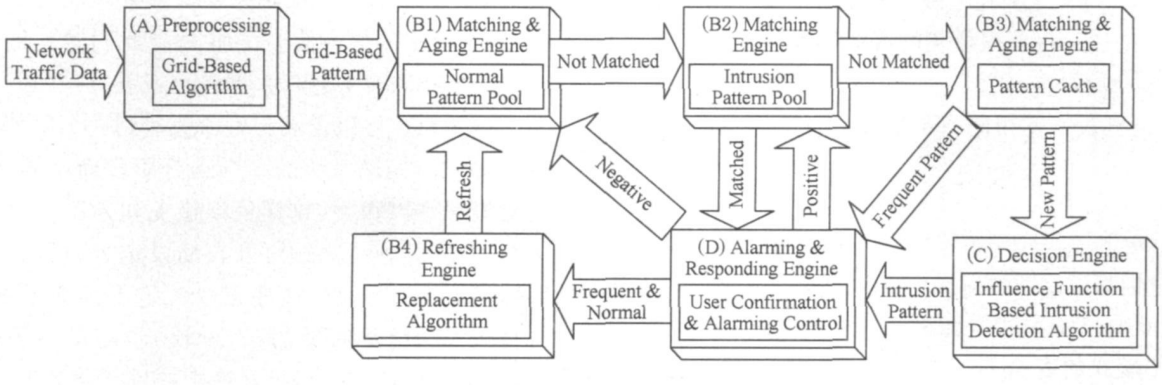

Fig. 1 Online adaptive network anomaly detection system model.

图 1 在线自适应网络异常检测系统模型

2.1 数据预处理

网络数据首先要经过预处理, 目的是将源数据转换为适当的粒度再输入系统. 我们使用了基于网格的方法划分数据空间, 这里的网格划分是指将数据空间的每一维划分区间, 从而将整个数据空间划分成数目有限的超级长方体, 并以网格为单位来判断落入网格的数据是否正常. 这样能够大大减小系统的运算复杂度和存储复杂度.

网格的形式化定义如下:

设 $A=\{A_1, A_2, \cdots, A_d\}$ 是一个有界属性集合, $U=A_1\times A_2\times \cdots\times A_d$ 是一个 d 维数据空间. $V=(v_1, v_2, \cdots, v_d)$ 是 U 中的一个 d 维数据, 其中 v_i 在 A_i 中取值. 通过将每一个属性维分割成 N 个区间, 我们把数据空间划分成互不相交的超级长方体.

一个网格 C 就是在各个维中, 分别取一个区间得到的超级长方体: $C=(c_1, c_2, \cdots, c_d)$. 其中 c_i 为符号维时是一个有效取值, 为数字维时是 A_i 中一个左闭右开区间: $c_i=[l_i, h_i)$. 我们说一个实例 $V=(v_1, v_2, \cdots, v_d)$ 投影到单元 $C=(c_1, c_2, \cdots, c_d)$, 即当: $v_i=c_i$ (当 v_i 是符号值), 或者 $l_i\leqslant v_i< h_i$ (当 v_i 是数字值).

网格的划分方法直接决定了系统的学习和分类能力. 针对不同的数据空间会有不同的划分方法, 由于本文要使用 KDD99 数据进行实验, 这里我们以网络连接数据为例说明数据空间的划分方法.

数据空间划分的关键是如何将每一维属性划分成离散的区间. 在网络连接数据中, 有取符号值的属性, 也有取数字值的属性. 对于符号值属性(如“protocol—type”)或仅取 0 和 1 的二进制属性(如“logged-in”), 我们将每一个不同的取值作为一个划分. 对于数值型的属性, 我们按照特征分为 2 类处理:

1) 属性值为一个百分数, 或者属性的取值是 512 以内的整数;

2) 属性值是大于 512 的整数.

对于第 1 种类型的数值属性, 我们可以简单地将其划分为 N 个等长的区间; 对于第 2 类属性, 等区间划分和基于密度的划分等都是不适合的, 因为这些属性虽然取值范围比较大, 但是多数实例的取值都集中在一个相对较小的区间内, 如“duration”.

对于这类属性, 使用等频装箱法将数据点均匀地分布在不同区间中看似比较合适, 但是在处理数据流的前提下, 这个方法很难有效实现. 为此我们使用了一种效果近似的函数转换法, 用一个 S 型函数将属性值转换到(0, 1)区间上, 再将转换结果平均

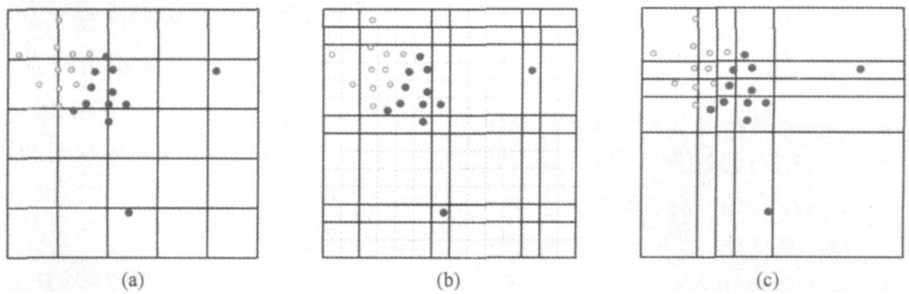

Fig. 2 Comparison of different discretization techniques. (a) Equal width; (b) Density based division; and (c) Variable transformation.

图 2 不同的网格划分方法比较. (a) 等分区间; (b) 基于密度划分; (c) 函数转换

分成 N 等份. 主要思想是在数据密度大的区域区间划分小一些, 在数据稀疏的区域区间划分大一些, 如图 2(c)所示.

这里我们使用的 S 函数如下:

$$f(x)=\frac{1}{1+e^{-(x-m)/c}}, \tag{1}$$

其中, m 和 c 代表当前属性历史数据的平均值和标准差. 它们都可以通过对历史数据的统计或相关的背景知识获取. 3.3 的实验证明了这种方式是有效的.

2.2 模式的表示

在本文中, 网络应用模式表示为一个四元组:

$$P=(C, R, H, L),$$

其中 C 是数据空间中的一个网格, R 是映射到网格 C 内的最近出现的 k 个实例集合, H 是此模式的生命值, L 是其标号.

在模式 P 中, R 是映射到 C 内的最近出现的 k 个网络实例. 保存这些实例的原因有 2 个, 一是当这个模式成为频繁模式后, 系统将向用户询问这一模式是否为正常, 网格这样的抽象表达方式用户是无法理解的, 这时可以列出这些实例让用户判断; 第 2 个原因是在系统进行增量学习时, 如果系统的网格划分方式有所调整, 可以通过这些实例将当前模式重新对应到正确的网格中, 从而保留知识.

H 是模式 P 的生命值, 当有新的实例投影到这个模式时, 其生命值会增加; 而长久没有实例匹配的模式, 其生命值会逐渐减小并最终被新的频繁模式替代. 通过对正常模式的这种运算可以使系统始终保持网络的最近工作状态.

L 是模式 P 的标号. 已询问过用户的模式, 标号是“正常”或“入侵”, 未确定的模式标号为“未标号”.

2.3 系统工作机制

在初始阶段, 所有的模式库都是空的, 决策模块中因为没有任何知识而无法工作. 所以在进行异常检测之前, 系统需要进行学习. 在学习阶段逐渐动态生成最近的正常模式库以及入侵模式库.

正常模式基本上都是频繁模式, 但是在实际情况中频繁模式并不都是正常模式. 例如, DoS 攻击包含大量网络流量, 也一定是频繁模式. 因此, 在学习过程中, 模块 B3 获得的频繁模式不能直接加入正常模式库, 而需要决策模块或用户的确认.

这里涉及到一个频繁模式的确定问题, 为了获得最近最经常出现的模式, 我们借鉴了操作系统中二级缓存的更新机制. 模式缓存库 PC 中每一个模式都有一个生命值 H, 当一条新记录匹配这个模式时, 这个模式的生命值会增加 1; 与此同时其他模式的生命值将减少 $\varepsilon(\varepsilon \ll 1)$, 即老化或衰减.

频繁模式可以定义为生命值达到一个阈值 τ 的模式. τ 的确定随系统的不同运行阶段而不同, 在初始阶段 τ 可以很大, 这样可以避免初始阶段对用户进行大量的询问. 比如可以自动设定为占所有输入数据的 5%以上的模式才可以认定为频繁模式, 或者每天只询问最频繁的 10 个模式. 随着系统的运行, 正常模式库和入侵模式库中的模式逐渐增多, 多数数据会在经过 B1 和 B2 时结束处理. 这时 τ 的值就可以根据流入 B3 的数据量的减少而自动减小, 不需用户调整.

频繁模式经过认定后会分别加入正常模式库 NPP 或入侵模式库 IPP. B1 和 B2 中模式库的大小可以不作限制, 随着模式的动态生成和衰亡, 模式库的大小会稳定在一定范围内. 但是如果内存有限需要进行限制时则要用到模式的更新策略, 这里简单地用新模式替换第 1 个生命值最小的旧模式. 同时, 这种模式的动态更新机制也使得系统可以适应网络使用环境的变化.

模块 C 是系统提供的一个开放平台, 此处可以集成多种检测算法, 甚至可以结合其他基于误用的检测系统, 为用户判断一个新模式的危险性提供参考信息. 这里我们给出了一个基于模式影响度的算法, 在第 2.4 节描述.

由于报警与响应不是本文的重点, 因此模块 D 的主要功能是回答询问和响应报警. 当然, 为了减轻用户的负担, 提高系统的可用性, 当模块 C 提供的参考信息具有较高的确信度时, 系统也可以根据这些信息自动进行认定.

2.4 基于模式影响度的检测算法

在网络异常检测的过程中, 我们依赖如下的假设: 正常数据之间或入侵数据之间具有一定的相似性, 而入侵数据与正常数据之间有一定的差异性. 这样每个数据对周围的数据都会有一个正面的或负面的影响. 一个新的待分类数据可以根据所有其他已分类数据对它的影响来决定其类型, 下面说明相关概念与方法.

2 个模式 P, Q 的距离 $D(P, Q)$ 定义为它们所包含的 2 个 d 维网格 C_P, C_Q 之间的距离:

$$D(C_P, C_Q)=\sum_{i=1}^{d} d(C_P^i, C_Q^i), \tag{2}$$

其中 $d(C_P^i, C_Q^i)=\begin{cases}0, & \text{if } C_P^i=C_Q^i;\\ 1, & \text{otherwise.}\end{cases}$

一个模式 P 对另一个模式 X 的影响函数定义为

$$f(P, X) = \exp(-D(P, X)). \quad (3)$$

这样, 当 $|NPP|>1$ 且 $|IPP|>1$ 时, 一个新模式 X 对正常模式的隶属度为

$$Fn(X) = \sum_{P\in NPP} f(P, X) / |NPP|. \quad (4)$$

X 对入侵模式的隶属度为

$$Fi(X) = \sum_{P\in IPP} f(P, X) / |IPP|. \quad (5)$$

X 的正常度定义为

$$N(X) = Fn(X) / Fi(X). \quad (6)$$

我们可以设置一个阈值 θ, 当 $N(X)>\theta$ 时我们认为新模式 X 为正常模式, 否则为异常模式并进行报警. 用户可以调整这个阈值以在高检测率和低误报率之间进行权衡. 检测算法如下所示.

算法 **1**. 基于影响度的网络异常检测算法.

输入: 模式衰减系数 ε; 频繁模式阈值 τ; 正常度阈值 θ.

初始化: 正常模式库、入侵模式库、模式缓存库初始都为空.

每当一个网络连接记录到达, 进行下列处理:

1) 按第 2.2 节网络应用模式的定义, 将此记录转化为模式 X;

2) 在正常模式库中搜索与 X 匹配的模式, 在搜索的同时累加所有正常模式对 X 的影响度值得到 Fn, 并对正常模式的生命值衰减 ε;

若发现与 X 匹配的模式, 则将其生命值加 1, 并结束对此记录的处理;

3) 在入侵模式库中搜索与 X 匹配的模式, 在搜索的同时累加所有入侵模式对 X 的影响度值得到 Fi;

若发现与 X 匹配的模式, 则报警, 并结束对此记录的处理;

4) 在模式缓存库中搜索与 X 匹配的模式, 在搜索的同时对缓存模式进行衰减;

若发现与 X 匹配的模式, 则将其生命值加 1, 若其生命值大于 τ, 则向管理员发出一个增加正常模式的申请, 并根据管理员反馈将 X 加入正常模式库, 结束对此记录的处理;

5) 此时 X 是一个新模式, 根据 Fn 和 Fi 计算其正常度, 若正常度小于等于 θ, 则报警, 并根据管理员反馈将 X 加入入侵模式库; 若正常度大于 θ, 则将 X 加入模式缓存库, 结束对此记录的处理.

在判断一个新的频繁模式时, 可能会遇到这个模式所保存的 k 个最近实例中既有正常连接又有异常连接的情况. 必要时我们可以将网格划分进行细化, 即在数据空间的某一维或几维上多一个区间分割点, 从而使这些冲突的实例被划分到不同的网格. 同时, 系统可以根据每个模式保存的 k 个实例为所有已存在的模式重新分派新的网格, 从而保留已经学习到的知识.

3 实验与结果分析

3.1 实验数据

实验使用 KDD99[12] 数据集. 它是 MIT Lincoln 实验室提供的 1998 DARPA 入侵检测评估数据集的一个扩充版本. 其中包括训练集(kddcup.data.gz)和测试集(corrected.gz). 数据以网络连接的形式保存, 每条记录含 42 个属性, 其中 7 个符号属性, 34 个数值属性, 1 个分类标号属性.

我们在实验中, 训练集主要使用了一个 10%的子集(kddcup.data－10－percent.gz). 其中共有数据 494020 条, 正常数据 97277 条, 入侵数据 396743 条, 入侵种类 22 种. 同时为了验证系统的可伸缩性, 我们也使用了训练集的全集进行了实验比较.

测试集则使用完整的 corrected.gz 数据集, 其中共有数据 311029 条, 正常数据 60593 条, 入侵数据 250436 条, 入侵种类 37 种, 其中有 17 种未在训练集中出现.

3.2 实验过程

系统用 Java 编写, 运行于一台 Intel Core Duo 2.4 GHz, 1GB 内存的电脑. 为了避免打开大文件所消耗的磁盘读取时间, 系统直接使用了训练数据和测试数据的压缩文件作为输入文件, 并在系统内部解压缩后进行处理. 训练数据集和测试数据集顺序一次性流过系统.

在训练集通过时, 系统处于“学习”工作方式, 在学习时系统并未用到所有训练数据的标号, 只是当需要用户确认一个频繁模式是否正常时系统会自动提取此模式包含的 k(实验时取 $k=10$)个最近训练数据的标号进行判断, 如果入侵数据占半数以上则认为此模式为入侵模式.

在测试集通过时, 系统可以分别处于“检测”和“检测时学习”2 种工作方式. 如果选择“检测时学习”, 系统发现新的频繁模式后会对比测试集上提供的标号来更新模式库. 下面的实验如无特别声明, 我们都是选择“检测”模式进行.

3.3 数据空间划分

在学习和检测之前, 首先要对数据空间进行网格划分. 我们使用第 2.1 节的方法, 符号属性每一个不同的取值划分一个区间; 数值属性划分成 N 个区间. 为了选择合适的 N, 我们测试了不同的取值, 并在划分结束后将 10%训练集的数据进行投影, 观察划分效果. 结果如表 1 所示, 其中 G 表示包含有实例的网格数, MG 表示其中既包含正常实例又包含入侵实例的网格数及其所占比例, FS 表示在 MG 中若按包含的实例投票表决确定网格的标号后被错判的实例总数及其在所有实例中的比例.

Table 1 Result of Data Space Gridding

表 1 网格划分效果

N	G	MG	FS
3	2615	44(1.68%)	427(0.086%)
4	5306	38(0.72%)	129(0.026%)
5	4675	37(0.79%)	190(0.038%)
6	6450	32(0.50%)	126(0.026%)
7	6322	44(0.70%)	176(0.036%)
8	10976	32(0.29%)	94(0.019%)
9	8615	49(0.57%)	185(0.037%)
10	10974	35(0.32%)	100(0.020%)
11	11462	40(0.35%)	163(0.033%)
12	14874	31(0.21%)	85(0.017%)

结果显示, 这种划分方法基本上能用较少的网格将正常数据和入侵数据有效地划分开. 根据训练数据集来看, 当 $N=8$ 时, 以网格为单位确定落入此区域的实例是否正常所带来的误差为 0.019%. 由于 KDD CUP'99 的获胜方法的检测率为 91.9%, 相比而言这个误差的数量级是可以接受的. 因此下面的实验我们选择参数 $N=8$ 对数据空间进行网格划分, 并以网格为学习和检测的基本单位.

3.4 实验结果与分析

实验主要考察系统的检测率和误报率:

检测率 DR(detection rate)=检测出的异常记录数/异常记录总数;

误报率 FPR(false positive rate)=判断为异常的正常记录数/判断为异常的所有记录数.

在测试时, 为了尽量保持学习时得到的知识, 我们取模式衰减参数 $\varepsilon=0.0001$, 模式库最大容量限制为 1000. 并为频繁模式阈值 τ 和正常度阈值 θ 选取了不同的值. 实验结果如图 3 所示:

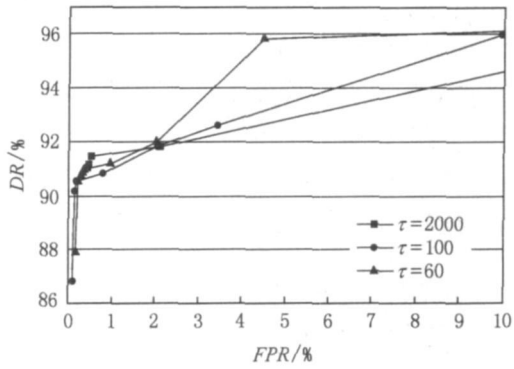

Fig. 3 ROC curves of experimental results.

图 3 实验结果的 ROC 图

可以看出, 当误报率控制在 2%以内时, τ 的取值对检测效果影响不大, 这说明系统检测能力主要是由少数频繁度比较高的模式决定的.

图 4 是当 $\tau=50$, $\theta=1.1$, $\varepsilon=0.0001$ 时, 模式库容量(PPS)取不同的值所获得的结果. 可见随着 PPS 的增加, 系统的误报率明显减小, 而当 PPS 超过 600 时, 系统性能趋于稳定.

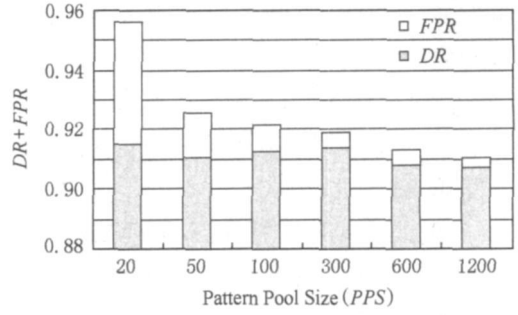

Fig. 4 Detection result vs. pattern pool size.

图 4 模式库容量对检测结果的影响

为了减小计算复杂度 PPS 不必太大, 但是也不能太小. 当我们将 PPS 减小到 20 以下时, 系统基本不能进行有效的检测了. PPS 为 300 时的检测结果列于表 2:

Table 2 Detection Results for *PPS*=300

表 2 ***PPS*=300** 时的检测结果 %

θ	DR	FPR
0.2	90.67	0.23
0.5	90.87	0.31
0.7	91.01	0.38
0.9	91.32	0.43
1	91.36	0.44
1.5	91.46	1.14
3	91.62	1.83

我们看到, 当 $\theta=0.9$ 时系统检测率为 91.32%, 误报率为 0.43%. 其对不同类型的攻击检测率如表 3 所示:

Table 3 Detection Results of Different Attack Types

表 3 对不同攻击类型的检测结果 %

Attack Type	*DR* of OANAD	*DR* of KDD 99 Winner
Probe	86.08	83.3
DoS	97.18	97.1
U2R	58.64	13.2
R2L	0.16	8.4

可见, 影响检测率的入侵类型主要是 U2R 和 R2L. 这主要是由于这 2 种入侵在训练集中出现较少(U2R 为 52 条, R2L 为 1126 条), 并且这 2 类入侵在行为模式上多数与正常数据也比较接近. 由于本系统是根据数据相似性来进行检测的, 因此会出现上述检测率较低的情况, 但是对于频繁出现的数据模式, 如 Probe 和 DoS 攻击, 系统有很高的识别率, 甚至优于 KDD CUP'99 获胜者. 作为对比, 表 4 列出其他方法在相同测试集上取得的结果.

Table 4 Comparison with Other Approaches

表 4 其他检测方法检测结果 %

Approaches	*DR*	*FPR*
KDD CUP'99 Winning entry	91.945	0.546
KDD CUP'99 Second place	91.525	0.576
Distributed learning[10]	91.7	3.2
Average GEdIDS[11]	90.581	0.565
Best GEdIDS-FP rate[11]	91.017	0.434

虽然 OANAD 的检测率不是最好, 但已接近 KDD CUP'99 获胜者的检测效果, 而误报率是最低的, 这点在异常检测中尤为重要. 如果综合考虑下列因素, 系统的性能是比较突出的.

1) 系统是轻量级的. 如果系统的正常模式库和入侵模式库的总容量是 M, 算法检测 n 条记录的时间复杂度为 $O(M\times n)$. KDD CUP'99 的获胜方法之一 MP13 使用 PERGAMENT software 运行了 6 h 完成全部计算, 而我们的算法只运行了不到 40 s. 其中学习 494020 条训练数据使用了不到23 s; 检测测试数据使用不到 17 s. 为了进一步测试系统的性能, 我们将训练集的全集输入系统进行学习, 结果系统只用了 277.5 s 就结束了学习, 其中还包括了解压缩的时间.

另外, 系统的空间复杂度低, 如果我们按 $PPS=300$ 计算, 系统最后共保留 600 个频繁模式(其中正常模式和入侵模式各 300 个), 只占用极少的内存空间.

2) 系统没有用到训练集的所有标号信息. 只是在分类频繁模式时查看了这些模式最近出现的 10 个实例进行判断, 仅占训练集标号的很少部分.

3) 系统是具有自学习能力的动态系统. 随着模式库的完善, 检测能力也会提高. 当我们选择"检测时学习"模式进行检测时, 系统在检测过程中增加了 64 个模式, 并达到检测率 94.12%和误报率 0.45% ($\tau=60$, $\theta=1.2$, $PPS=1000$)的结果.

4) 检测新入侵的能力. corrected.gz 测试集含有入侵种类 37 种, 其中有 17 种未在训练集中出现, 当允许误报率为 3.40%时本系统可以检测到所有 17 种入侵; 在误报率为 0.33%时可检测到 11 种. 表 5 列出了其捕获的 11 种新入侵记录数.

Table 5 Detection Ability of New Type of Intrusions

表 5 对新入侵的检测结果

New Intrusions	#Total	#Detected
apache2	794	267
httptunnel	158	110
mailbomb	5000	0
mscan	1053	826
named	17	1
processtable	759	418
ps	16	5
saint	736	707
sendmail	17	1
snmpgetattack	7741	0
snmpguess	2406	3
sqlattack	2	1
udpstorm	2	0
worm	2	0
xlock	9	0
xsnoop	4	2
xterm	13	0

4 总 结

本文提出了一个在线自适应网络异常检测系统模型, 它不需要特殊的训练集, 它的学习模式和工作模式是统一的, 能够在使用的过程中逐步学习用户的正常模式, 并在每一次与用户的交流中确认入侵模式, 修正检测依据. 实验结果表明, 系统效率很高, 具有较好的检测率和满意的误报率.

参 考 文 献

[1] Heberlein L, Dias G V, Levitt K N, et al. A network security monitor[C] //Proc of the 1990 Symp on Security and Privacy. Los Alamitos, CA: IEEE Computer Society, 1990: 296-304

[2] Callegari C, Vaton S, Paqano M. A new statistical approach to network anomaly detection [C] //Proc of the 2008 Int Symp on SPECTS. Los Alamitos, CA: IEEE Computer Society, 2008: 441-447

[3] Tian Xinguang, Gao Lizhi, Sun Chunlai, et al. Anomaly detection of program behaviors based on system calls and homogeneous Markov chain models[J]. Journal of Computer Research and Development, 2007, 44(9): 1538-1544 (in Chinese)
(田新广, 高立志, 孙春来, 等. 基于系统调用和齐次Markov链模型的程序行为异常检测[J]. 计算机研究与发展, 2007, 44(9): 1538-1544)

[4] Fava D, Byers S, Yang S. Projecting cyberattacks through variable-length Markov models [J]. IEEE Trans on Information Forensics and Security, 2008, 3(3): 359-369

[5] Duc D, Matsumoto K, Takishima Y, et al. Two-stage incremental working set selection for fast support vector training on large datasets [C] //Proc of the 2008 IEEE Int Conf on RIVF. Los Alamitos, CA: IEEE Computer Society, 2008: 221-226

[6] Latifur K, Awad M, Thuraisingham B. A new intrusion detection system using support vector machines and hierarchical clustering [J]. The VLDB Journal, 2007, 16(4): 507-521

[7] Mao Guojun, Zong Dongjun. An intrusion detection model based on mining multi-dimension data streams[J]. Journal of Computer Research and Development, 2009, 46(4): 602-609 (in Chinese)
(毛国君, 宗东军, 基于多维数据流挖掘技术的入侵检测模型与算法[J]. 计算机研究与发展, 2009, 46(4): 602-609)

[8] Yasami Y, Khorsandi S, Mozaffari S, et al. An unsupervised network anomaly detection approach by *k*-means clustering & ID3 algorithm[C] //Proc of the 2008 IEEE Symp on ISCC. Los Alamitos, CA: IEEE Computer Society, 2008: 398-403

[9] Dartique C, Jang H, Zeng W. A new data-mining based approach for network intrusion detection[C] //Proc of the 7th Annual Conf on CNSR. Los Alamitos, CA: IEEE Computer Society, 2009: 372-377

[10] Liu Yanheng, Tian Daxin, Yu Xuegang, et la. Large-scale network intrusion detection algorithm based on distributed learning[J]. Journal of Software, 2008, 19(4): 993-1003 (in Chinese)
(刘衍珩, 田大新, 余雪岗, 等. 基于分布式学习的大规模网络入侵检测算法[J]. 软件学报, 2008, 19(4): 993-1003)

[11] Folino G, Pizzuti C, Spezzano G. GP ensemble for distributed intrusion detection systems[C] //Proc of the 3rd Int Conf on Advanced in Pattern Recognition. Berlin: Springer, 2005: 54-62

[12] ACM. KDD Cup 1999 Data[OL]. [2001-06-30]. http://www.sigkdd.org/kddcup/

Wei Xiaotao, born in 1971. PhD candidate. His main research interests include data mining and network security.
魏小涛, 1971 年生, 博士研究生, 主要研究方向为数据挖掘和计算机网络安全.

Huang Houkuan, born in 1940. Professor and PhD supervisor. Senior member of China Computer Federation. His main research fields include artificial intelligence, data mining, and machine learning.
黄厚宽, 1940 年生, 教授, 博士生导师, 中国计算机学会高级会员, 主要研究方向为人工智能、数据挖掘、机器学习等.

Tian Shengfeng, born in 1944. Professor and PhD supervisor. His main research interests include artificial intelligence and network security.
田盛丰, 1944 年生, 教授, 博士生导师, 主要研究方向为人工智能和网络安全.

Research Background

With the extensive usage of computer networks, security becomes a critical issue. Network intrusions can cause severe disruption to networks. Therefore there is an urgent need for a solution that can actively defend networks against the growing security threats. The intrusion detection systems(IDS) can automatically scan network activity and recognize intrusion attacks to protect computers against unauthorized uses and make them secure and resistant to intruders. This is where network IDS comes in to offer security in addition to that provided by traditional anti-threat applications such as firewalls, antivirus software and spy-ware detection software. From the last decade, misuse detection has been the dominant strategy for IDSs for the reasons that it is easier to implement. However, anomaly detection has the advantage of detecting novel intrusions without any prior knowledge. This research presents an online adaptive network anomaly detection system. It runs in real time and dynamically maintains its knowledge base. The experimental results shows that this light weighted system achieves a relatively high detection rate and very low false positive rate. This research work is supported by the National Natural Science Foundation of China under grant No. 60442002.