一种无监督的数据库用户行为异常检测方法

李海斌[1],李 琦[1],汤汝鸣[1],吴 珺[1],吕志远[1],裴 丹[1],史俊杰[2],董 旭[2],房双德[2],杨一飞[2],吴 烨[2]

[1](清华大学 计算机科学与技术系,北京 100084)
[2](百度公司,北京 100085)
E-mail:lihb15@tsinghua.org.cn

摘 要: 检测数据库内部合法用户的异常行为,对防范内部攻击和数据泄露具有重要意义,然而面临如下挑战:攻击模式不确定,真实异常样例少,数据集缺少准确标注.人工设定阈值和规则难以有效应对复杂多样的异常.本文提出了一种基于无监督学习的用户行为异常检测方法,通过划定时间窗口统计提取特征,运用核密度估计算法分别从单维度、多维度建模,实现在海量的无标注历史日志中发现简单异常和复杂异常、在新的线上数据中检测异常.真实数据实验表明,该方法能够有效检测出简单异常,实验中检测三种简单异常的平均严格查准率和宽松查准率分别达 90% 和 100%;能够从多维度找出存在攻击嫌疑的复杂异常,实验中成功检测出了一种单维度无法检测出的新的复杂异常.

关 键 词: 无监督学习;数据库;用户行为;异常检测;内部数据泄露
中图分类号: TP309 **文献标识码:** A **文 章 编 号:** 1000-1220(2018)11-2464-09

User Behavior Anomaly Detection for Database Based on Unsupervised Learning

LI Hai-bin[1],LI Qi[1],TANG Ru-ming[1],WU Jun[1],LV Zhi-yuan[1],PEI Dan[1],SHI Jun-jie[2],DONG Xu[2],FANG Shuang-de[2],YANG Yi-fei[2],WU Ye[2]

[1](Department of Computer Science and Technology,Tsinghua University,Beijing 100084,China)
[2](Baidu,Beijing 100085,China)

Abstract: Detecting anomalous behaviors of legal internal users is of great significance for guarding against insider attack and preventing data leakage. However,challenges exist:uncertainty of attack strategies,lack of real anomaly cases and accurately labeled data,etc. Human-defined thresholds and basic rules are not enough for detecting complex and various anomalies. This paper proposes a method for user behavior anomaly detection based on unsupervised learning,aiming to effectively discover both simple and complex anomalies in unlabeled massive history log and detect anomalous behavior in new online data,by applying Kernel Density Estimation to single dimension and multiple dimensions respectively. Experiments on real data show that this method can detect simple anomalies in single dimension with an average strict precision of 90% and relaxed precision of 100%,and successfully discover a hidden complex anomaly in multiple dimensions which can not be detected by single dimension approach.

Key words: unsupervised learning;database;user behavior;anomaly detection;insider data leakage

1 引 言

数据库存储着企业的大量敏感数据,有的甚至涉及金融、财务、电子商务、在线支付等核心机密数据,因此防范数据泄露、确保数据安全是企业十分迫切的安全需求.一旦发生数据泄露的安全事故,将给企业造成无法挽回的损失.数据泄露可以由外部攻击或内部攻击造成,而由内部用户发起内部攻击导致的数据泄露后果尤为严重.由于内部用户对数据(包括密级数据)拥有合法的访问权限,对整个系统非常熟悉,如果其恶意滥用权限、偷取数据、泄露数据,将造成更致命的攻击和更严重的损失[3,4].因此,检测数据库内部合法用户的异常行为具有重要意义.

数据库用户行为异常检测主要面临三大挑战:

1)攻击模式不确定.为了规避检测,内部攻击者可能会将其恶意的行为隐藏在正常的行为中,不易发现;攻击策略没有固定模式,无法事先预知;

2)真实异常样例少.由于内部员工叛变或者黑客冒充合法用户攻击是小概率事件,可获取的真实异常事件样本极少;正、负样本数量极不平衡,历史日志中绝大部分都是正常行为的数据,异常数据所占比例很小,不足以准确描绘异常的特征,无法直接从数据中学到所有异常的特性;

3)人工标注十分困难.真实环境中采集的数据通常没有

收稿日期:2018-01-26 收修改稿日期:2018-04-03 基金项目:国家自然科学基金项目(61472214)资助. 作者简介:**李海斌**,男,1988 年生,硕士研究生,CCF 会员,研究方向为智能运维和大数据安全;**李 琦**,男,1979 年生,博士,副研究员,CCF 高级会员,研究方向为移动安全和大数据安全;**汤汝鸣**,男,1991 年生,博士研究生,研究方向为智能运维;**吴 珺**,女,1993 年生,硕士研究生,研究方向为网络异常检测;**吕志远**,男,1991 年生,博士研究生,研究方向为网络安全;**裴 丹**,男,1973 年生,博士,副教授,博士生导师,CCF 会员,研究方向为智能运维;**史俊杰**,男,1990 年生,硕士,高级工程师,研究方向为大数据安全和隐私保护;**董 旭**,男,1987 年生,架构师,研究方向为大数据与追踪取证技术;**房双德**,男,1986 年生,博士,资深工程师,研究方向为机器学习和数据安全;**杨一飞**,男,1983 年生,硕士,高级架构师,研究方向为数据安全;**吴 烨**,男,1973 年生,博士,主任架构师,研究方向为网络安全与隐私保护.

准确的“正常”或“异常”标注,要将海量无标注的历史日志转化为有准确标注的数据集是非常困难的,需要非常丰富的领域知识和大量的时间、人力投入. 而且,这类标注比图像分类等问题的标注更加复杂,因为用户行为的异常具有多样性、抽象性,甚至还有一些是已有知识无法覆盖的未知异常,导致人对异常的认识本身存在局限性,难以准确标注.

现有的数据库安全防护策略(或工具)难以有效防范内部攻击. 入侵检测系统和防火墙,主要用于防范来自外部的攻击,难以防范内部合法用户发起的攻击[5]. 常用的数据库安全机制,如用户标识与鉴权、基于角色的访问控制、数据加密[6]等,虽然能够防范用户访问未授权数据,但无法防范合法用户对已授权数据的滥用和泄露. 基于机器学习的异常检测方法分为有监督和无监督两种,有监督的方法依赖于准确标注好的数据集进行模型训练,需要充足的正、负样本,然而由于存在着如前所述的“三大挑战”,有监督的方法无法发挥作用,因此,本文采用无监督的方法来进行异常检测.

本文提出了一种基于核密度估计算法[1,2]的无监督机器学习方法,对数据库用户行为建模、检测异常. 该方法主要有三个步骤:首先通过特征工程从原始日志中提取特征;然后采用核密度估计算法分别从单维度和多维度训练得到某个用户(或用户群体)正常行为的概率密度模型,计算合理的概率密度阈值;最后在检测阶段,根据样本的概率密度是否低于阈值来判定该样本是否为偏离用户绝大多数正常行为的离群点,从而检测异常、发出告警.

本文提出的方法具有以下优点:

1)无需对历史数据进行费时费力的人工标注,能够在正负例不平衡(正例数量远大于负例数量)、真实异常样例缺失、数据集缺少标注或完全无标注的情况下进行异常检测;

2)无需对数据的分布进行任何预先的假设,直接通过核密度度估计算法从历史数据中无监督地训练得到其概率密度分布曲线;

3)能够对不同的用户(或用户群体)分别训练模型,实现个性化检测,且通过定期重新训练模型即可实现阈值的自动更新. 在真实数据上的实验表明,该方法具有良好的异常检测效果,检测出的样本绝大部分都是具有攻击嫌疑或高风险威胁的异常;从单维度检测简单异常的严格查准率和宽松查准率分别为90%和100%,从多维度检成功测出了一种单维度无法发现的新的复杂异常,验证了该方法能够有效辅助安全运维人员缩小排查范围、锁定重点异常、聚焦安全威胁,提升安全工作效率.

本文的框架如下:第二章介绍相关工作,第三章介绍方案设计,第四章详细介绍特征工程模块,第五章详细阐述基于核密度估计算法的无监督异常检测模型,第六章给出实验评估结果并进行讨论,第七章进行总结.

2　相关工作

关于数据库用户行为异常检测国内外已有一些相关研究,主要包括:基于参考阈值的方法[9]、基于关联规则挖掘的方法[21]、基于免疫原理的方法[20]、基于隐 Markov 模型的方法[19]、基于聚类的方法[22]、基于模式挖掘的方法[23,24]、基于query 语句向量化特征的方法[5,13-15]、针对数据库应用层检测的方法等[11,12].

基于参考阈值的方法由 Spalka[9]等人提出,该文献设计了一套数据库管理系统的误用检测系统,对比了基于参考阈值和基于“Δ-关系”的两种检测方法,然而这两种方法都需要大量的计算开销,而且无法区分不同用户的异常行为. 实际生产环境中,人工设定阈值的方法也能够检测一些简单的异常,然而存在不足:指标数量多,对人的领域知识要求高;阈值难以设准,人工更新阈值缺乏可靠依据;检测能力有限,只能检测已经发生过的异常或攻击. 文献[21]研究了一种基于关联规则挖掘的方法,将聚类和关联挖掘技术相结合,设计相应的异常检测系统框架. Chung[10]等人设计了用于关系型数据库误用检测的 DEMIDS 系统,采用频繁项集来表征多次使用的 query 语句. 文献[20]研究的是基于免疫原理的方法,提出了一种结合免疫原理的数据库入侵检测技术. 文献[19]和文献[22]分别研究了基于隐 Markov 模型和基于聚类模型的异常检测方法. 文献[23]和文献[24]研究了基于模式挖掘的用户行为异常检测方法(主要针对恶意终端用户行为的检测). Bertino[13]等人、Ashish Kamra[14]等人、Shebaro[15]等人和 Sallam[5]等人研究了基于 query 语句向量化特征的检测方法,特征提取时,将用户每一条 query 语句分别提取生成一个样本,即每一种操作、每一个表名、每一个列名分别对应一个特征,这种方法将导致特征的维度会随着数据库、数据表、列的数量扩大而增加,不适用于数据表项庞大的数据库,特别是当数据规模较大时,特征维度急剧增长将带来额外的开销. Lee[11]等人和 Hussain[12]等人分别提出了 DIDAFIT 系统和 DetAnom 系统,这两种方法都是在数据库的应用层进行异常检测,不能够区分不同用户的异常行为.

与这些相关的工作不同,本文设计了一套新的特征工程方案,通过划定时间窗口统计特征,提取出的特征能够反映数据库用户的行为习惯和特点,而且其中包含了直接与数据库常见攻击场景相对应的重要特征指标,同时还确保了特征维度保持稳定、不会随着数据库规模的扩大而增长;在模型训练时,本文不对数据的分布作任何假设,而是用核密度估计算法从数据中无监督地训练得到其历史分布规律;此外,本文将数据库的异常分为简单异常和复杂异常,分别从单维度和多维度建立模型同时对这两类异常进行检测,并在真实生产环境的数据集上验证了方法的有效性.

3　方案设计

3.1　问题定义

本文将数据库用户行为“异常”定义如下:当用户的操作行为严重偏离其绝大多数正常历史行为的轨迹时,或者当用户的操作行为具有拖取数据、滥用数据等攻击嫌疑,对数据库的安全构成威胁时,称为“异常”.

假设绝大部分正常行为的数据服从某个未知的分布 X,用函数 $y=f(x)$ 表示其未知的概率密度函数,采集的历史数据集 D 可以认为是由分布 X 中独立抽取一定数量的样本构成的,那么用户行为异常检测需要解决的问题,等价于判断给定的任意样本是否属于分布 X. 若属于分布 X,则判定该样本

为正常,若不属于 X,则该样本为异常.

根据异常的复杂程度、检测难易程度,本文把异常分为两类:简单异常和复杂异常.简单异常指通过单维度分析就能够直接判定的异常,这类异常通常对应了一些简单的攻击场景,主要表现为单个重要特征指标上出现的偏离绝大多数正常样本的离群值.比如大量数据拖取异常,只需要对“访问数据量”这个特征进行单维度分析就可以检测出来.复杂异常指的是通过单维度分析无法判定的异常,其表现形式更加复杂,需要将多个特征指标组合起来分析才能发现其“异常之处”,比如本文在实验中检测出的“24 小时连续访问”异常,或其它一些事先未知的新异常.

3.2 基本思路

本文进行异常检测的总体思路是:从海量无标注的历史日志中提取特征,运用无监督学习算法训练得到用户(或用户群体)绝大部分正常行为的概率密度分布,若被检测样本的概率密度低于概率密度判定阈值,则判定为异常.该方法基于数据库异常检测场景的两个基本前提:一方面,异常行为在某些属性上总是偏离绝大多数正常行为的;另一方面,真实的异常攻击是罕见的,只占很小的比例,在对正常行为建模时,历史数据中包含的极少量异常样本对模型的影响微乎其微.本文把数据库历史数据中异常样本所占的比例称为可疑异常率.可疑异常率是一个很小的数值,由负责安全运维的人员根据历史发生异常事件的频率和安全需求的严格程度综合评估后设定.根据可疑异常率,可以找出概率密度最低的部分样本上报为异常.

本文的异常检测系统框架如图 1 所示.从要素上看,系统包含两个最核心的模块:特征工程模块和机器学习模型训练

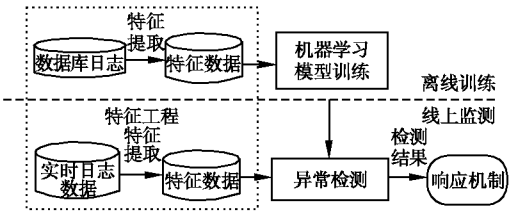

图 1 异常检测系统框架

Fig. 1 Overall design of anomaly detection system

模块.特征工程实现从原始日志数据到可供机器学习算法直接使用的特征数据的转化.模型训练模块将无监督机器学习算法运用于特征数据,从而训练得到单维度和多维度的异常检测模型,分别用于检测简单异常和复杂异常.从流程上看,系统可以分为离线训练阶段和线上检测阶段.离线训练时,从海量无标注的数据库历史日志中提取特征,而后采用无监督机器学习算法在不同用户(或用户群体)的特征数据上分别进行模型训练,得到该用户(或用户群体)对应的单维度模型和多维度模型.线上检测时,从线上日志中提取特征,再用离线训练得到的模型同时进行简单异常和复杂异常的检测.若检测出异常,则系统发出告警,触发相应的响应机制,包括由安全运维人员进行人工排查、调查取证、对存在安全隐患的不当行为进行提醒警告或对证据确凿的攻击行为进行追责处理等.

接下来,将对特征工程、模型训练这两个核心模块进行详细介绍,并给出真实数据上的实验评估结果.

4 特征工程

本文针对数据库用户行为异常检测设计了一套全新的特征工程方案,实现将海量、杂乱、无标注的原始日志转化为可供机器学习算法使用的特征数据.杂乱的原始日志中含有大量有价值的信息,包括用户基本信息、数据库信息(如数据库名、表名、属性名)、操作信息(如查询、添加、删除等完整 SQL 语句)、时间戳信息等,这些信息需要通过特征工程从原始日志中提取、转化为有效的特征,从而用于后续的模型训练.本章分别从特征的选取、特征的提取流程两个方面来介绍特征工程方案.

4.1 特征选取

结合日志信息的完备程度,综合考虑潜在异常或攻击行为的表现,本文选取了一系列能够反映用户行为的特征(见表 1),主要有以下两类:

1)用户属性相关的特征,比如用户角色(用户组)、用户工作状态(包括在岗、离职审批中、已离职等)、用户访问位置(单日访问发起时所在不同客户端的个数).

表 1 实验中选取的特征

Table 1 Features selected for experiment

编号	特征名称	统计提取粒度	对应攻击场景(简单异常)
1	用户角色	N/A	N/A
2	用户工作状态	N/A	N/A
3	用户发起访问位置	天	N/A
4	访问数据量	天、两小时	大量数据拖取
5	访问不同表总个数	天、两小时	多表大范围访问
6	发起鉴权请求失败比例	天、两小时	访问控制漏洞嗅探
7	访问绝密级列比例	天、两小时	绝密级敏感数据拖取

2)操作行为的统计特征,本文将用户执行的 query 语句按时间窗口聚合,统计得到操作行为的相关特征.在选取特征时,安全运维人员根据经验对数据库可能面临的攻击场景(例如表 1 列举的 4 种攻击场景)进行了假设和预估,通过分析这些场景下攻击行为的表现,有针对性地选定对应的特征指标.此外,考虑到用户在某个时刻的孤立行为有时不足以反映出任何异常的特性,例如,多次、少量的数据拖取在某一个具体的时刻可能看不出异常之处,但在较长时间后会有一定累积,通过特定长度时间窗口上的总吞吐量上观测可以更容易发现这种异常的行为,因此本文提出了按时间窗口聚合统计的策略,实验中采用了“天”和“两小时”两种时间粒度.

本文实验中选用的操作行为相关的特征包括:单日内访问数据量(对应大量数据拖取异常)、单日内访问不同表总个数(对应多表大范围访问异常)、单日内发起的鉴权请求失败占当日总请求的比例(对应访问控制漏洞嗅探异常)、单日内访问绝密级别的列占所有访问列的比例(对应绝密级敏感数据拖取异常).为了反映用户一天内操作行为的时间分布,对于这些与特定攻击(异常)场景相对应的重要特征指标,除了按天统计外还按两小时的粒度做了进一步的统计提取,相当于把按“天”统计的这些特征进一步分解为 12 个按“两小时”统计的特征.

在实际应用中,特征的选取应当根据业务本身的特点和

采集日志信息的完备程度来确定,无需局限于表 1 中列举的特征. 例如,若日志记录的信息更加完备,还可考虑提取如下特征:用户单日新建库和新建表类型操作比例、删除库和删除表类型操作比例、含“select ＊”语句比例等.

4.2　提取流程

本文特征提取的流程主要包括以下几步:将原始的数据库日志按用户(或用户群体)分类归档、按时间窗口聚合、统计特征数值、标准化处理.

按用户(或用户群体)分类归档. 采集到原始日志后,首先将历史日志按照用户(或用户群体)分类归档,后续的特征提取和模型训练都将在每个用户(或用户群体)各自对应的日志数据上分别进行.

按时间窗口聚合、统计特征数值. 统计的时间窗口粒度可以根据实际情况灵活选择,细粒度的时间窗口可以选取分钟、小时为单位,粗粒度的可以选取天或是周为单位,便于观察用户行为的时间规律和周期特征. 本文在实验中主要以“天”为单位提取特征,即用户一天内所有的日志数据提取生成一个样本,同时,还将表 1 中编号 4-7 的这四个重要的特征分解为 12 个按“两小时”统计的细粒度特征.

标准化处理. 由于在不同的特征维度上的度量单位有所不同,在进行模型训练之前需当将特征数据进行标准化处理,即对每一个维度分别求出均值和方差,并将该维度上的原始数据减去均值后再除以方差.

综上,本文选取与用户属性和操作行为密切相关的统计值作为特征,采取了按“天”、“两小时”聚合统计的策略,将用户每一天的日志数据分别提取生成一个训练样本. 该方法在数据库规模扩大、用户数量增多、数据表项急剧增长的情况下依然适用,提取样本的特征维度能够保持稳定,能够有效控制特征复杂度和运算开销,有助于提高模型训练效率,缩短分析、检测时间.

5　无监督异常检测模型

为应对数据集无标注的挑战,本文采用无监督机器学习方法进行模型训练,其中算法选用核密度估计算法. 针对可能同时存在的简单异常和复杂异常,本文从单维度和多维度分别建立概率密度模型,即建立“单维度概率密度模型”检测简单异常、同时建立“多维度概率密度模型”检测复杂异常,从而实现对两类异常同时检测. 本章将分别介绍核密度估计算法、单维度概率密度模型和多维度概率密度模型.

5.1　核密度估计算法

核密度估计是一种用于估计随机变量未知概率密度函数的非参数检验方法[1,2]. 假设从概率密度为未知函数 f 的某个分布中独立抽样 n 个样本($x_1, x_2, x_3, \cdots, x_n$),则由公式(1)可以求得该分布概率密度函数 f 的估计值:

$$f_b(x) = \frac{1}{n}\sum_{i=1}^{n} K_b(x - x_i) = \frac{1}{nb}\sum_{i=1}^{n} K\left(\frac{x - x_i}{b}\right) \quad (1)$$

其中 K 为核函数,是积分值为 1 的非负函数;b 为带宽,是一个平滑参数. 核密度估计算法可以简单地理解为将每个样本作为中心点对应的核函数的加权求和.

由于本文关注的是正常行为的整体分布情况,考虑到高斯核函数能够使估计的分布更加平滑,因此实验中核函数 K 选用高斯核函数.

带宽 b 的取值能够直接影响到密度估计的效果[1,2],如果 b 值选得过大,得到的概率密度估计曲线将过于平滑,从而无法反映出局部区间的变化和差异;如果 b 值选得过小,得到的概率密度估计曲线将过拟合,即在曲线中出现过多、过于细微的波动. 参数 b 有多种调参方法[17,18],本文在实验中采用了 python 的超参自动搜索模块 GridSearchCV 对参数 b 进行调优[7,8].

5.2　单维度概率密度模型

单维度概率密度模型用于检测单维度简单异常. 简单异常通常是某些重要特征指标在单维度上的离群值,一般为该维度上概率很小、数值“极大”的值. 例如,大量数据拖取、多表大范围访问、绝密敏感数据拖取这三种简单异常,它们可以分别对应到以下这三个特征上的“极大值”:单日内访问数据量、访问数据表个数、访问绝密级列比例.

单维度概率密度模型训练时使用单个特征指标对应的单维度历史数据作为训练集,用核密度估计算法训练得出该维度的概率密度分布曲线,通过找出概率密度极低且数值极大的少数离群样本,实现检测对应简单异常的目的. 单维度概率密度判定阈值 S_0 可以根据可疑异常率 a 求出. 例如,若假定数据库历史数据中只有 1% 的概率出现某种简单异常,即 a = 1%,计算所有历史样本在该维度的概率密度值并做升序排列后,则排在前 1% 位置的样本对应的概率密度即为相应的概率密度判定阈值 S_0. 检测时,计算新样本在该维度的概率密度 P,若 $P <= S_0$,则判定该样本为该种简单异常,触发相应的单维度异常告警,进入人工排查和取证追责流程;若 $P > S_0$,则判定该样本不属于该种简单异常.

5.3　多维度概率密度模型

多维度概率密度模型用于检测复杂异常. 与单维度概率密度模型不同,多维度概率密度模型能够将多个维度关联起来分析,考察样本在高维空间的概率密度分布情况;通过找出高维空间中分布稀疏的样本,从而发现一些潜在的复杂异常,甚至可能是从未察觉的复杂异常.

设样本第 i 个特征(即第 i 个维度)的概率密度函数为 $d_i(x)$,样本的维度数为 m,则该样本的整体概率密度值 D 可由公式(2)求得:

$$D(x) \approx \prod_{i=1}^{m} d_i(x) \quad (2)$$

本文把通过公式(2)建立的概率密度模型称为多维度概率密度模型. 严格地说,当选取的 m 个特征之间满足条件独立时,样本的整体概率密度等于各个单维度的概率密度值相乘[16]. 由于本文选取的特征彼此之间没有明显的依赖关系,可以采用公式(2)近似求解样本的整体概率密度,而且这种近似计算的方法更加简单、开销更小,有助于快速找出高维空间中概率密度极小的样本,实验中取得了不错的效果.

模型训练时,在对每个特征建立单维度概率密度模型的基础上,多维度概率密度模型可以直接通过公式(2)求得. 与单维度模型检测异常的方法相似,多维度概率密度判定阈值 M_0 可以同理求得:设可疑异常率为 a,计算所有历史样本的多维度概率密度值并做升序排列后,排在前 a 位置的样本对应的概率密度值即为多维度复杂异常的概率密度判定阈值

M_0. 检测时,计算新样本的多维度概率密度值 P,若 P < = M_0,则判定该样本为多维度异常,触发多维度异常告警,进入人工排查、取证追责流程;若 P > M_0,则判定该样本不属于多维度异常.

6 实验与分析

本文的实验包括两部分:从单维度检测简单异常和从多维度检测复杂异常. 本章首先介绍实验环境和评估指标,而后给出了单维度和多维度两类异常检测实验的分析过程、检测结果和效果评估,最后进行了补充讨论.

6.1 实验环境

本文以某大型互联网公司的一个重要业务数据库(数据仓库)作为实验场景. 该数据库上线半年左右,共有 87 名用户(均为具有合法访问权限的内部员工),暂时没有靠人工方法抓获过真实内部攻击的异常案例,且采集的数据集完全没有标注. 考虑到该数据库历史真实攻击事件发生的概率极低、安全要求较高,经安全运维人员根据历史观察和运维经验综合评估后,将可疑异常率设定为 1%.

经特征工程,从历史日志提取了 1973 个样本. 单个用户对应样本数量 CDF 图如图 2 所示,样本数最多的用户拥有 168 个样本,最少的只有 1 个,平均每个用户拥有 22 个样本. 也就是说,在近半年的 180 天中,活跃天数最多的用户有 168 天的数据访问记录,最不活跃的用户只有 1 天的数据访问记录,平均每个用户有 22 天的活跃天数.

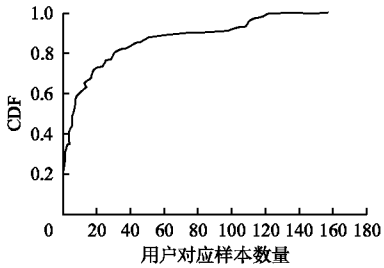

图 2 用户对应样本数量 CDF

Fig. 2 CDF of number of samples for user

经统计,如果把平均每 2 个工作日中至少有 1 天数据访问的用户称为高活跃度用户,把平均每月的访问天数少于 1 天的用户称为低活跃度用户,把介于低活跃度和高活跃度用户之间的其他用户称为普通活跃度用户的话,三类用户的分布为:高活跃度用户占 11%,普通活跃度用户占 46%,低活跃度用户占 43%. 高活跃度用户主要是负责日常业务的核心骨干,有比较频繁的访问操作,有较多的训练样本;而其他用户对数据库的访问频率较低,能够提取的样本数很少,有的甚至只有 1 个样本. 对样本数量较少的用户,暂时不宜单独建模. 为了对无法单独建模的用户也进行异常检测,本文提出把所有用户看成一个整体(即整个用户群体)的建模策略,即用所有用户的数据建模,得到适用于所有用户的模型. 为了表示区分,本文把使用单个用户的历史数据训练得到的模型称为单用户模型,把使用所有用户的历史数据训练得到的模型称为整体用户模型.

考虑到该数据库处于运营初期,样本规模较小,因此本文在实验中以整体用户模型来验证方法的有效性,即以所有用户的 1973 个样本作为整体进行训练,分别建立整体用户的单维度和多维度概率密度模型,同时进行单维度和多维度异常检测. 关于整体用户模型和单用户模型的适用场景和运用方法将在 6.5 小节专门讨论. 下文中将以整体用户模型为例,给出实验的分析过程和评估方法.

6.2 评估指标

本文通过对模型检测出的样本进行人工逐个排查,结合用户的访问详情(如 SQL 上下文,时间地点,频率等),核实样本是否为具有攻击嫌疑或高风险威胁、值得重点关注的真实异常,从而求出模型检测异常的查准率,来衡量模型是否能够帮助安全运维人员发现和检测出有价值的异常.

在模型报出的样本集合中,若将被安全运维人员排查认定为具有攻击嫌疑或高风险威胁的真实异常的样本数量设为 TP(True Positive),被安全运维人员认定为不具有攻击嫌疑、被模型误报为异常的样本数量设为 FP(False Positive),则查准率 Precision 可以由公式(3)求得. 将被模型漏报(即实际为真实异常)的样本的个数设为 FN(False Negative),则查全率 Recall 可以由公式(4)求得. 然而,由于在真实的环境中数据集完全没有标注(无 ground truth),且无法准确掌握训练数据中存在哪些真实异常、是否有未知的异常,导致 FN 无法准确评估,查全率 Recall 无法通过公式(4)准确求得. 因此,本文在实验中主要以 Precision 作为评估指标,着重评估模型从无标注数据集中发现和检测出有价值异常的能力.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

此外,经排查发现,被报出的异常在显著程度上是有差别的,因此本文将具有明显攻击嫌疑或高风险威胁、且概率密度极低的样本认定为“强显著”异常,具有潜在的攻击嫌疑或风险威胁、且概率密度低的样本认定为“弱显著”异常,把概率密度低但不具有攻击嫌疑和风险威胁的样本认定为正常. 若只考虑“强显著”异常,求得的查准率本文称为“严格查准率”;若把“强显著”异常和“弱显著”异常同时考虑,求得的查准率本文称为“宽松查准率”.

由于相关工作中其它方法适用的数据集和本文完全无标注的数据集有所区别,而且本文采用的是全新设计的特征工程方案,并首次提出将异常分为简单异常和复杂异常、分别从单维度和多维度进行检测,因此在评估效果时无法直接和其它方法进行比较. 故本文在评估时着重考察本方法在完全无标注的真实环境数据集上检测异常的效果,评判该方法是否能够满足实际生产环境的要求、达到预期的效果.

6.3 单维度异常检测实验分析和效果评估

本节针对三种常见的简单异常,分别对整体用户建立了三个单维度概率密度模型,并对这三种简单异常的检测效果进行了评估.

6.3.1 单维度实验 1:大量数据拖取异常检测

大量数据拖取异常对应的特征为“访问数据量”,将该特征对应的单维度历史数据作为训练集,用核密度估计算法建立单维度概率密度模型,得到在标准化后的区间上的概率密

度分布曲线如图 3 所示. 从图 3 可以看出,在标准化的区间上该特征的概率密度模型呈三峰分布(其中两个是很小的波峰),主要集中于(-0.7,0)、(0.8,1.5)、(2.1,3.1)三个区间上,对应的概率密度峰值分别为 3.65、0.08 和 0.15.

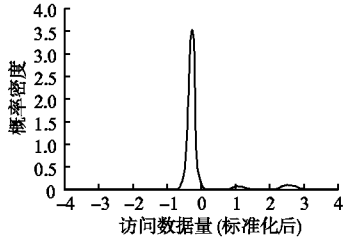

图 3　“访问数据量”单维度概率密度分布曲线

Fig. 3　Density curve of feature “Data Volume”

通过“访问数据量”单维度概率密度模型,求得历史数据 1973 个样本的概率密度值,对应 CDF 图如图 4 所示,90% 的样本概率密度值集中于区间(3.58,3.72),只有少量的样本(小于 10%)集中于极低的(0,0.2)的区间. 单维度概率密度值越低,意味着样本在该维度上的离群程度越高.

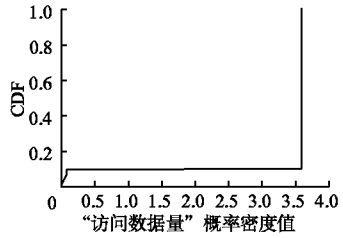

图 4　“访问数据量”概率密度累积分布函数

Fig. 4　CDF of “Data Volume” density

由于可疑异常率 a = 1%,则 1973 个历史样本中约有 20 个可疑的大量数据拖取异常. 将所有样本的概率密度值升序排列,最小的 20 个样本即为可疑的异常,上报排查. 排在第 20 位的样本对应的概率密度值即为大量数据拖取异常的概率密度判定阈值 S_0. 通过计算,得 $S_0 = 0.0063$. 检测时,若新样本在该维度的概率密度值大于 0.0063,则判定为正常;若小于 0.0063,则发出告警、上报为可疑的大量数据拖取异常,并进入人工排查、取证追责流程.

6.3.2　单维度实验 2:多表大范围访问异常检测

多表大范围访问异常对应的特征为“访问不同表总个数”,其单维度概率密度模型在标准化区间上的分布曲线如

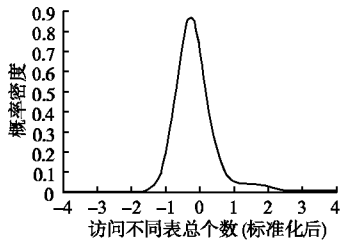

图 5　“访问不同表总个数”单维度概率密度分布曲线

Fig. 5　Density curve of “Table Total”

图 5 所示. 与“访问数据量”的曲线不同,此概率密度分布曲线呈单峰分布,与高斯分布的形状有些相似.

“访问不同表总个数”单维度概率密度累积分布函数如图 6 所示,约有 85% 的样本概率密度值集中于区间(0.83,0.86),只有极少量的样本(约占 5%)概率密度值集中于极低的(0,0.05)区间. 类似地,根据可疑异常率,计算得概率密度判定阈值 $S_0 = 0.0051$. 检测时,若新样本在该维度的概率密度值大于 0.0051,则判定为正常;若小于 0.0051,则发出告警、上报为可疑的多表大范围访问异常,并进入人工排查、取证追责流程.

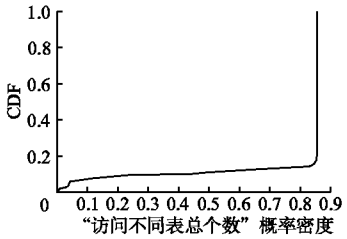

图 6　“访问不同表总个数”概率密度累积分布函数

Fig. 6　CDF of “Table Total” density

6.3.3　单维度实验 3:绝密级敏感数据拖取异常检测

绝密级敏感数据拖取异常对应的特征为“访问绝密级列比例”,其单维度概率密度模型在标准化区间上的分布曲线如图 7 所示. 从图 7 可以看出,该特征的概率密度曲线在(0.2,4.0)区呈微小的波浪状分布.

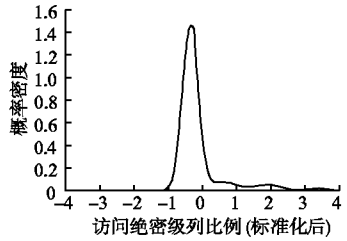

图 7　“访问绝密级列比例”单维度概率密度分布曲线

Fig. 7　Density curve of “Top Secret Column Ratio”

该特征单维度概率密度累积分布函数如图 8 所示,约有 86% 的样本概率密度集中于区间(1.42,1.44),只有极少量的样本概率密度值较低. 根据可疑异常率,计算得概率密度判定

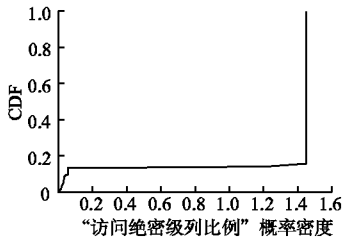

图 8　“访问绝密级列比例”概率密度累积分布函数

Fig. 8　CDF of “Top Secret Column Ratio” density

阈值 $S_0 = 0.0176$. 检测时,若新样本在该维度的概率密度值大于 0.0176,则判定为正常;若小于 0.0176,则发出告警、上报为可疑的绝密级敏感数据拖取异常,并进入人工排查、取证追责流程.

6.3.4 单维度三种简单异常的检测效果评估

按1%的可疑异常率,三种简单异常共检测出60个可疑的样本(每种20个).由安全运维人员对这60个可疑异常进行逐个人工排查,得到检测的"严格查准率"和"宽松查准率"如表2所示.实验表明,三种简单异常的严格查准率平均为90%,宽松查准率平均为100%;在配置为24核Intel(R)Xeon(R)CPU E5-2620 v3,2.40GHz,64GB RAM的服务器上,模型检测单个新样本耗时平均为0.549ms

表2 检测三种简单异常的时间性能和查准率

Table 2 Time and Precision of simple anomaly detection

项目	单样本检测耗时	严格查准率	宽松查准率
大量数据拖取	0.532 ms	95%	100%
多表大范围访问	0.578 ms	100%	100%
绝密敏感数据访问	0.539 ms	75%	100%
平均	0.549 ms	90%	100%

经人工排查,单维度模型检测出的三种异常样本在该维度的特征数值100%具有以下两个特点:

1)概率密度值极低;

2)对应的原始数值绝大部分属于或接近该维度的极大值.

基于这两点,被检测出的样本至少是"弱显著"异常,甚至还有可能是风险程度更高的异常.以多表大范围访问异常为例,概率密度最低的20个样本全部属于"访问不同表总个数"单维度top 20的极大值,意味着这些样本属于十分罕见、且存在特大范围数据表访问行为的异常.

通过结合用户访问详情(如SQL上下文、时间地点、频率等)进一步排查分析发现,模型检测出的三种简单异常样本中包含"强显著"异常的比例分别占95%、100%和75%.例如,在大量数据拖取异常样本中,排查发现对应的用户还存在多次、多表单日全量数据无"limit"拖取等高风险操作行为;在多表大范围访问异常样本中,还排查发现对应用户存在大量的"load数据使用"、"insert overwrite"等大量导入导出的高风险行为等,这些均表明样本属于具有明显的攻击嫌疑或高风险威胁的"强显著"异常.

综上,单维度模型检测简单异常取得了良好的效果,能够从单维度快速、准确找出有价值的异常,帮助安全运维人员从重要的单维度指标中快速筛选出具有攻击嫌疑和高风险威胁的样本,为启动进一步的风险应急处置流程提供了线索和依据,同时,能够有效辅助安全人员有针对性地聚焦于重要指标存在的威胁和隐患,集中精力对模型检测出的重要异常事件采取更加深入细致的安全措施,包括对相关用户进行调查询问、收集证据、提醒警告、追责处理等.

6.4 多维度异常检测实验分析和评估

本节对整体用户建立多维度概率密度模型进行检测实验,并对检测效果进行评估.

通过建立多维度概率密度模型,由公式(2)求得所有历史样本的整体概率密度值,得到如图9所示的CDF图.从图中可以看出,历史数据的整体概率密度值(取对数)大部分集中在区间(-20,7).根据实验中选取的可疑异常率1%,求出的概率密度判定阈值 M_0 =6.23E-52.检测时,若新样本的整体概率密度值大于 M_0,则判定为正常,若小于 M_0,则判定为多维度可疑的异常.

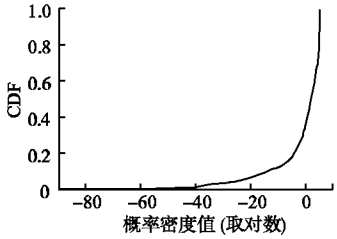

图9 多维度概率密度值(取对数)的累积分布函数

Fig. 9 CDF of multi-dimension log density

为了评估多维度模型检测异常的效果,本文对检测出的多维度可疑异常样本进行了逐一人工排查,发现整体概率密度值最低的这20个样本中,19个样本属于"24小时连续访问"异常,1个样本属于"频繁鉴权失败"异常.其中,"频繁鉴权失败"异常,属于通过单维度也能够检测出来的简单异常,而"24小时连续访问"异常是单维度无法发现的新的复杂异常.

这20个样本当天内每两小时访问表的数量如表3所示,

表3 多维度可疑异常样本当天内每2小时访问表的数量

Table 3 Table access in every 2 hours of suspicious samples

序号 \ 时间 / 表数	0-2	2-4	4-6	……	18-20	20-22	22-24
1	1554	1373	1174	……	1199	1754	1084
2	2417	708	1504	……	801	917	1192
3	2048	307	180	……	951	2943	2214
4	1019	542	1057	……	380	137	195
5	610	507	900	……	617	741	913
6	473	389	439	……	1313	1239	1096
7	498	260	392	……	393	222	202
8	795	539	515	……	279	443	187
9	521	181	173	……	721	268	226
10	1817	683	612	……	131	125	70
11	719	310	203	……	1102	1072	799
12	542	148	334	……	559	230	228
13	417	320	216	……	851	265	181
14	526	445	333	……	706	736	523
15	789	0	0	……	1244	1777	1064
16	2645	412	134	……	306	159	291
17	379	397	474	……	494	144	242
18	382	275	246	……	2337	1142	1480
19	1006	646	612	……	34	50	33
20	437	259	328	……	288	353	400

第一列表示当天凌晨0点至2点访问表的数量,其它列以此类推,可以看出,这20个样本中除了第15个样本外,其它样本在一天内的任意两小时内访问表的数量均大于0,意味着这些样本在当日内对数据库的操作基本没有间断.通常情况下,用户一天内的访问行为是有一定规律的,主要的访问量集中于白天,在下班休息时间一般没有或者极少出现访问行为,深夜加班或临时任务等特殊情况通常少见.即使是喜欢在夜间工作的"夜猫子",一天内至少会有适当的休息时间.因此,24小时连续访问是一种非常可疑的、少见的异常行为.如果这些异常的背后不是人为操作,而是自动化的程序在访问数据库,那么这种异常和真实的攻击已经十分接近,存在极大的

安全风险、将可能导致极为严重的危害.因此,安全运维人员认定这 19 个“24 小时连续访问”样本具有极大的攻击嫌疑和极高的风险威胁,属于“强显著”的异常,并进行了更加深入的二次调查.

经过详细的调查取证后确认,这 19 个“24 小时连续访问”强显著异常是由某平台管理员采用自动化程序进行数据的导入、导出和格式转换等工作所导致.结合该用户的访问详情,并与其它业务行为关联分析后最终确认,该用户的行为没有恶意的动机,不属于恶意的攻击.然而,通过模型检测出的“24 小时连续访问”异常本身具有非常大的意义,而且非常有必要.

此外,多维度异常检测找出的这 19 个“24 小时连续访问”异常样本,从单维度看不出异常的迹象,只有通过多维度关联起来综合分析,才能发现其异常的物理意义.而且在进行检测实验之前,这类异常完全没有被察觉和关注,也没有预先设定任何规则对这类异常进行防御,因此,这类异常属于多维度模型找出的一种新的“复杂异常”.

综上,多维度检测实验中,若只考虑检测复杂异常的效果,严格查准率和宽松查准率均为 95%,而且发现的是一种之前未被关注的新的“强显著”复杂异常;若把实验中检测出的 1 个简单异常样本也考虑在内,则检测异常的严格查准率和宽松查准率均为 100%,即检测出的所有异常样本均为具有攻击嫌疑或高风险威胁的、值得重点关注的异常.

6.5 讨　论

本节主要对整体用户模型和单用户模型的适用场景和运用方法进行说明.

整体用户模型和单用户模型的原理和方法是相通的,区别在于整体用户模型的训练数据是所有用户的样本,而单用户模型的训练数据是单个用户的样本.从背后的含义看,单用户模型检测的离群样本是偏离该单个用户行为的异常点,而整体用户的模型检测出的离群样本是偏离所有用户行为的异常点,相比之下,整体用户的模型检测出的样本是“离群”的性质更加严重、级别更高的异常.

在实际生产环境中,用户数据量不均衡,特别是在数据库运营初期,单个用户训练样本规模太小,不适宜采用单用户模型检测异常.针对此,本文提出了在此阶段可首先建立适用于所有用户的“整体用户模型”,从而实现在运营初期对无法单独建模的用户也能够进行异常检测.当数据库运营较长时间后,单用户样本数量将有一定的积累,而且随着业务周期的变化,一些不活跃的用户也可能转变为活跃用户,等单个用户的训练样本数量充足时,可再建立单用户模型进行更高精度、更个性化的检测,从而实现整体用户模型和单用户模型同时检测、互相补充.

7 结　语

检测数据库内部合法用户的异常行为,面临许多挑战:攻击模式不确定,真实异常样例缺失,数据集缺少准确标注等.针对这些挑战,本文提出了一种基于无监督学习的用户行为异常检测方法,设计了一套专门的特征工程方案,运用核密度估计算法分别从单维度、多维度建立概率密度模型,实现在海量无标注历史日志和新生成的线上日志中发现和检测异常的用户行为.通过某互联网公司的真实数据实验表明,本文提出的方法能够从繁杂的数据库历史日志中有效检测出具有攻击嫌疑或高风险威胁的异常样本,检测简单异常的严格查准率和宽松查准率分别达 90% 和 100%,并成功检测出了一种之前未被安全运维人员注意到的新的复杂异常,能够有效辅助安全运维人员锁定重点异常、聚焦安全威胁,提升安全工作的效率,为应对和防范数据库的安全威胁提供了一套可行的方案.

References:

[1] Rosenblatt M. Remarks on some nonparametric estimates of a density function[J]. Annals of Mathematical Statistics,1956,27(3):832-837.

[2] Wertz W. Density estimation for statistics and data analysis-B. W. Silverman[M]. Berlin Heidelberg,Metrika,1988:58-59.

[3] Legg P A,Buckley O,Goldsmith M,et al. Automated insider threat detection system using user and role-based profile assessment[J]. IEEE Systems Journal,2017,11(2):503-512.

[4] Legg P A,Buckley O,Goldsmith M,et al. Caught in the act of an insider attack:detection and assessment of insider threat[C]. In:IEEE International Symposium on Technologies for Homeland Security,2015:1-6.

[5] Sallam A,Bertino E,Hussain S R,et al. DBSAFE—an anomaly detection system to protect databases from exfiltration attempts[J]. IEEE Systems Journal,2017,11(2):483-493.

[6] Wu Pu-feng, Zhang Yu-qing. An overview of database security [J]. Computer Engineering,2006,32(12):85-88.

[7] Pedregosa F, Gramfort A, Michel V, et al. Scikit-learn: machine learning in python[J]. Journal of Machine Learning Research, 2013,12(10):2825-2830.

[8] Buitinck L, Louppe G, Blondel M, et al. API design for machine learning software:experiences from the scikit-learn project[C]. In: European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases Workshop on Languages for Machine Learning,2013:1-15.

[9] Spalka A,Lehnhardt J. A comprehensive approach to anomaly detection in relational databases[C]. In:Working Conference on Data and Applications Security,2005:207-221.

[10] Chung, Yip C,Gertz,et al. DEMIDS:a misuse detection system for database systems[C]. In:Integrity and Internal Control in Information Systems,2000:159-178.

[11] Lee S Y,Low W L,Pei Y W. Learning fingerprints for a database intrusion detection system[C]. In:European Symposium on Research in Computer Security,2002:264-280.

[12] Hussain S R,Sallam A M,Bertino E. detAnom:detecting anomalous database transactions by insiders[C]. In:ACM Conference on Data and Application Security and Privacy,2015:25-35.

[13] Bertino E,Terzi E,Kamra A,et al. Intrusion detection in RBAC-administered databases[C]. In: Proceedings of the 21st Annual Computer Security Applications Conference,2005:1-20.

[14] Ashish Kamra,Evimaria Terzi,Elisa Bertino. Detecting anomalous access patterns in relational databases [J]. The International Journal on Very Large Data Bases,2008,17(5):1063-1077.

[15] Shebaro B,Sallam A,Kamra A,et al. PostgreSQL anomalous query detector[C]. In:International Conference on Extending Data-

base Technology,2013:741-744.
[16] Chaitali Gupta, Ranjan Sinha, Yong Zhang. Eagle: user profile-based anomaly detection for securing Hadoop clusters[C]. In: IEEE International Conference on Big Data,2015:1336-1343.
[17] Duong T. Ks: kernel density estimation and kernel discriminant analysis for multivariate data in R[J]. Journal of Statistical Software 2007,21(7):1-16.
[18] Sheather S J,Jones M C. A reliable data-based bandwidth selection method for kernel density estimation[J]. Journal of the Royal Statistical Society,1991,53(3):683-690.
[19] Kuang Zhu-fang, Yang Guo-gui, Li Qing, et al. Hidden-markov-model-based anomalous detection techniques for database systems [J]. Journal of Computer Research and Development, 2006, 43 (z3):257-261.
[20] Li Xiao-hua, Dong Xiao-mei, Yu Ge. Research on database intrusion detection technology based on immune theory[J]. Journal of Chinese Computer Systems,2009,30(12):2343-2347.
[21] Kuang Zhu-fang, Tan Jun-shan. KMApriori: an Efficient anomalous detection approach to database systems[J]. Computer Engineering & Science,2008,30(6):18-21.
[22] Zhong Yong, Lin Dong-mei, Qin Xiao-lin. An Algorithm of unsupervised anomaly detection based on DBMS and its application [J]. Computer Science,2007,34(1):123-127.
[23] Lian Yi-feng, Dai Ying-xia, Wang Hang. Anomaly detection of user behaviors based on profile mining[J]. Chinese Journal of Computers,2002,25(3):325-330.
[24] Song Hai-tao, Wei Da-wei, Tang Guang-ming, et al. Anomaly detection of single user behaviors based on pattern ming[J]. Journal of Chinese Computer Systems,2016,37(2):221-226.

附中文参考文献:

[6] 吴溥峰,张玉清. 数据库安全综述[J]. 计算机工程,2006,32(12):85-88.
[19] 邝祝芳,阳国贵,李 清. 基于隐 Markov 模型的数据库异常检测技术[J]. 计算机研究与发展,2006,43(z3):257-261.
[20] 李晓华,董晓梅,于 戈. 基于免疫原理的数据库入侵检测方法研究[J]. 小型微型计算机系统,2009,30(12):2343-2347.
[21] 邝祝芳,谭骏珊. KMApriori:一种有效的数据库异常检测方法[J]. 计算机工程与科学,2008,30(6):18-21.
[22] 钟 勇,林冬梅,秦小麟. 一种基于 DBMS 的无监督异常检测算法及其应用[J]. 计算机科学,2007,34(1):123-127.
[23] 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.
[24] 宋海涛,韦大伟,汤光明,等. 基于模式挖掘的用户行为异常检测算法[J]. 小型微型计算机系统,2016,37(2):221-226.