doi:10.11959/j.issn.1000-436x.2017142

# Hadoop 云平台用户动态访问控制模型

杨宏宇，孟令现
（中国民航大学计算机科学与技术学院，天津 300300）

**摘 要**：为解决 Hadoop 云平台无法动态控制用户访问请求的问题，提出一种基于用户行为评估的 Hadoop 云平台动态访问控制（DACUBA, dynamic access control based on user behavior assessment）模型。该模型首先实时收集用户指令序列，通过并行指令序列学习（PCSL, parallel command sequence learning）获取用户行为轮廓。然后利用前向轮廓建立全局 *K* 模型，对后续行为序列进行分类并对分类结果进行评估。随后将评估结果与改进 Hadoop 访问控制机制结合，使云平台用户的访问权限随自身行为动态改变。最后通过实验验证了模型算法的有效性和动态访问控制机制的可行性。

**关键词**：云平台；Hadoop；用户行为；访问控制；并行指令序列学习
**中图分类号**：TP393 **文献标识码**：A

# Hadoop cloud platform user dynamic access control model

YANG Hong-yu, MENG Ling-xian
( School of Computer Science and Technology, Civil Aviation University of China, Tianjin 300300, China)

**Abstract:** In order to solve the problem that Hadoop cloud platform could not dynamically control user access request, a Hadoop cloud dynamic access control model based on user behavior assessment (DACUBA) was proposed. The model first collected the user instruction sequence in real time and the user behavior contour was obtained by parallel command sequence learning (PCSL). Then the global *K* model was established by using the forward profile, the subsequent sequence was classified and the classification results were evaluated. The evaluation results were combined with the improved Hadoop access control mechanism to make the cloud platform users' access rights change dynamically with their own behaviors. Experimental results demonstrate that the model algorithm is effective and the dynamic access control mechanism is feasible.

**Key words:** cloud platform, Hadoop, user behavior, access control, parallel command sequence learning

## 1 引言

Hadoop 作为开源分布式计算云平台，因其特有的高可靠性、高扩展性、高效性和高容错性等优点，得到了各大电商及互联网企业的广泛应用[1]，与此同时，其安全问题也日益突出[2]。在众多的云安全问题中，数据安全是云安全的核心问题之一。访问控制通过限制用户对数据信息的访问能力及范围，从而保证资源不被非法使用和访问，成为云平台中数据安全的重要保障。而现有 Hadoop 云平台在安全访问控制机制设计上并没有充分考虑其用户正常或异常的属性变化，使其存在重大安全隐患。

目前，国内外针对云平台安全机制的研究取得了一定的进展。Gupta 等[2]基于密度估计和 PCA 主成分分析方法设计了一种 Hadoop 平台异常检测系统，实时监控 Hadoop 平台用户行为，由于该方法缺乏相应的容错机制和异常用户处理机制，反而会增加云平台管理员的工作量。Tan 等[3]提出一种基

**收稿日期**：2016-12-13；**修回日期**：2017-04-12
**基金项目**：国家科技重大专项基金资助项目（No.2012ZX03002002）；中国民航科技基金资助项目（No.MHRD201009, No.MHRD201205）
**Foundation Items:** The National Science and Technology Major Project (No.2012ZX03002002), The Science & Technology Project of CAAC (No.MHRD201009, No.MHRD201205)

于信任度的动态访问控制模型，但没有将信任模型与访问控制模型很好地结合在一起，且仅进行了理论上的分析。Jing 等[4]提出了一种基于用户行为评估的云平台动态访问控制模型，由于没有描述用户行为的检测方法且其模型过于复杂，故不能较好地与现有 Hadoop 云平台结合。

针对以上问题，本文提出一种新的基于用户行为评估的动态访问控制模型。该模型通过收集 Hadoop 平台用户行为序列，采用并行指令序列学习算法从中提取用户行为模式，利用全局 $K$ 模型对用户行为进行分类，将用户行为分类结果进行行为评估，并将得出的评估值用于平台的访问控制机制中，从而提升了 Hadoop 云平台的安全性。

## 2 云平台动态访问控制模型

### 2.1 研究范围

本文研究的 Hadoop 集群拓扑如图 1 所示。KDC（key distribution center）为 Kerberos 密钥分配中心。

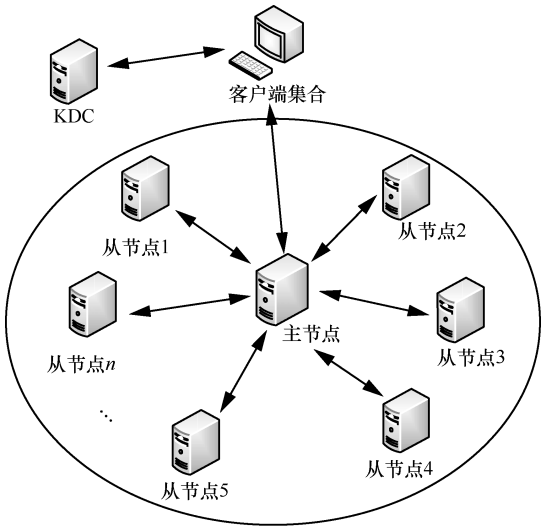


图 1 Hadoop 集群拓扑

在 Hadoop 集群中，存在以下 2 种类型的用户异常行为威胁平台安全：恶意用户窃取 Hadoop 云平台内部用户 session 或账号通过客户端登录到平台，进而对平台资源进行非法利用或破坏的异常行为；内部用户自身进行某些非法操作并试图越权访问他人数据或恶意消耗平台资源的异常行为。

本文研究目标主要针对以上 2 种异常情况，对于用户注册初始阶段进行恶意操作的行为，可以通过引入初期行为审查机制予以避免，该情况不属于本文研究范围。

### 2.2 设计思想

无论是恶意用户冒用还是内部用户中途进行恶意操作均会导致其行为轮廓的改变，由此可以根据当前行为轮廓与其相应的前向行为轮廓的变化程度来判断用户是否进行了异常操作，进而基于用户的异常操作的数量及异常程度对用户的行为进行评估，利用评估结果动态地控制用户对云平台的访问请求。DACUBA 模型的基本设计思想是：实时监控用户行为轮廓，利用用户前向轮廓进行判别、评估，进而动态地控制用户的访问。

结合 Hadoop 集群分布式拓扑结构和 DACUBA 模型的设计思想，设计动态访问控制流程如下。

1) 在主节点运行用户指令收集模块，用于收集所有用户在主节点的操作记录(*UserId*, *Cmd*, *Time*)，实现对所有用户行为的实时监控，其中，*UserId* 为用户 ID 号，*Cmd* 为命令，*Time* 为命令提交时间。

2) 主节点为每个用户建立行为数据库，存储用户行为数据，用于后续分析。

3) 主节点利用 Hadoop MapReduce 框架将用户行为数据分割分配到各个从节点计算，然后利用 Reduce 结果在主节点为每个用户单独建立 $K$ 个行为模式库，对其后续行为进行检测分类，进而利用行为评估公式计算出每个用户的评估值并存入评估值数据库。

4) 主节点根据每个用户的评估值给予相应的用户权限，当用户请求服务时，控制用户的访问行为，实现对用户的动态的访问控制机制。

### 2.3 动态访问控制模型

当前 Hadoop 云平台访问控制可分为 2 级：1)系统级访问控制(Service Level Authorization)，用于决定用户能否使用 Hadoop 云平台中的指定云服务；2) 用户操作级访问控制，包括 DFS Permmisions 和 Access Control on Job Queues[5]。这 2 级访问控制均采用静态访问控制策略，一旦授予用户相应的权限，则该用户将永久性地具有该权限直至管理员进行相应的权限更改，由此可见，该策略存在严重的安全性问题。在这2级访问控制中，系统级控制是最基础的访问控制，优先级高于用户操作级控制，因此，本文考虑在系统级控制层嵌入本文的动态访问控制模型，达到增加 Hadoop 云平台安全性的目的。

在结合现有 Hadoop 平台访问控制模型特点基础上，本文针对 Hadoop 平台设计的 DACUBA 模型如图 2 所示。

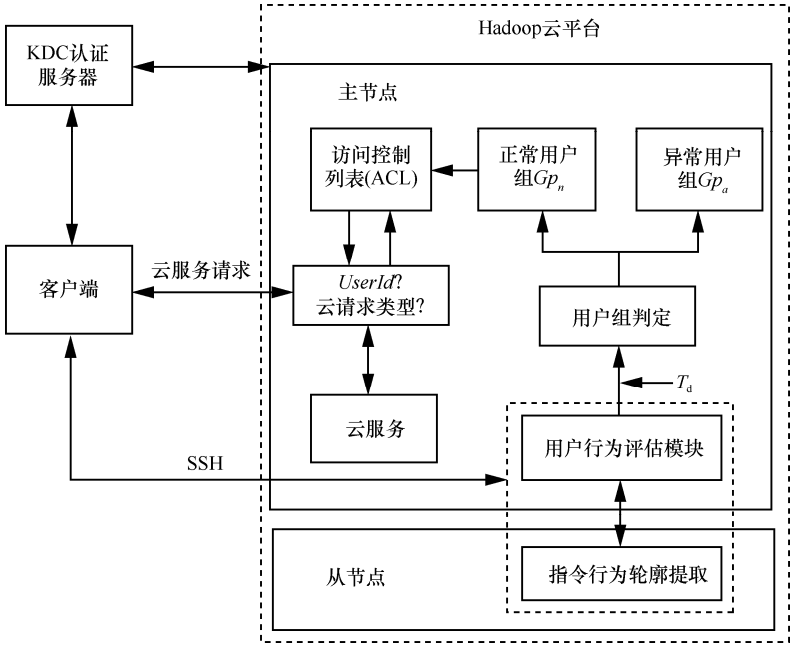


图 2　Hadoop 平台 DACUBA 动态访问控制模型

该模型包括用户行为评估模块、动态访问控制策略模块和 Hadoop 本身的 Kerberos 安全认证机制等核心模块。上述核心模块的功能设计如下。

1) 用户行为评估模块对 Hadoop 集群中得到 KDC 服务票据的用户进行指令序列收集并进行指令行为分类，利用行为评估公式对含有分类标签的行为序列进行行为评估，记录综合评估值。

2) 动态访问控制策略模块利用用户行为评估模块得到的用户综合评估值决定相应用户的访问权限。

3) Kerberos 认证机制提供集群内部认证通信方式并使用 KDC 服务器负责认证客户端和发放票据。

在 DACUBA 模型中，动态访问控制策略模块在传统的 RBAC 基础上结合现有 Hadoop 平台访问控制机制进行设计，引入了用户行为综合评估值的概念，访问控制策略中基本元素描述如下。

用户集 $U=(User_1, User_2, \cdots, User_i)$表示可以访问 Hadoop 云平台资源的所有主体的集合；角色集 $R=(R_1, R_2, \cdots, R_i)$，不同角色拥有云平台资源访问权限不同；权限集 $P=(P_1, P_2, \cdots, P_i)$表示用户主体通过角色访问云平台中客体资源（即访问对象，包括 HDFS 文件及 MapReduce 任务）的操作权限集合，与访问操作 HDFS 读、写、执行及 MapReduce 任务提交和任务撤销相对应，其操作权限集包含读写权限、执行权限、任务提交及撤销权限；用户行为综合评估值 $T(User_i)$表示系统根据某用户的行为进行综合评估后所给出的评估分数。

Hadoop 云平台的动态访问控制具体实现过程如下。

1) 角色划分。根据权限集 $P$ 中权限的不同组合划分角色。

2) 平台用户统一管理。将 Linux 系统用户映射为 Hadoop 用户，实现平台用户的统一管理。

3) 用户组建立。建立正常用户组 $Gp_n$ 和异常用户组 $Gp_a$，同时初始将所有通过 KDC 身份认证的用户加入 $Gp_n$ 并将 $Gp_n$ 添加到 Service Level Authorization ACL 中。

4) 用户分组策略。用户行为评估模块对平台用户 $User_i$ 行为进行实时分类评估，主节点将 $User_i$ 综合行为评估值 $T$ 同阈值 $T_d$ 进行比较：$T \leqslant T_d$，判断 $Gp_a$ 中有无 $User_i$，若无 $User_i$，则将 $User_i(UserId, Time_s, Time_v)$加入异常用户组 $Gp_a$，并记录加入时间的 $Time_s$ 及有效期限 $Time_v$，删除 $Gp_n$ 中的 $User_i$；若有 $User_i$，则重置其有效期限 $Time_v$。当 $Time_v \leqslant 0$，将 $User_i$ 重新添加入 $Gp_n$。

5) 访问请求响应。用户 $User_i$ 通过 KDC 认证服务器身份认证和主节点验证后向主节点发出云服务请求时，主节点据 Service Level Authorization ACL 中用户列表判断是否响应该请求：若 $User_i$ 在 ACL 列表中，则响应该请求，结合管理员授予该用

户的权限，为其分配不同的角色，实现对资源的访问；否则，通过 Token 返回拒绝标识并给出拒绝服务提示。

## 3 用户行为评估

DACUBA 模型中，用户行为评估模块起到了决定性的作用：用户行为综合评估值将直接影响平台用户的分组情况。因此，评估方法的合理性显得尤为重要。

评估方法的合理性需要体现在以下几个方面。

1) 用户行为证据的选取应能够正确反映用户的行为。

2) 用户行为分类的准确率高。

3) 用户近期行为的权重应大于历史行为的权重。

4) 用户的正常行为对用户行为评估值起正向作用，异常行为起负向作用。

5) 用户的异常行为对综合评估值的负向影响应随着重复的次数而急剧增加。

6) 用户正常行为对用户综合评估值提高的影响应远小于异常行为对其评估值降低的影响。

7) 用户综合评估值还应包含用户初始评估值和其他云平台平均推荐值等因素。

### 3.1 Hodoop 平台用户行为评估

在众多的 Hadoop 平台用户行为证据中，Hadoop 平台用户指令序列具有重复性和规律性，能够从中提取用户指令序列模式作为用户行为轮廓。当用户进行异常操作时，其行为轮廓与正常行为轮廓相悖，从而可以被识别分类。另外，与 CPU、内存、磁盘使用率等证据[6]相比，用户指令序列较易于收集。因此，本文选取 Hadoop 平台用户指令序列作为用户行为证据，采用文献[7]的 LZW 算法和字典压缩算法对用户指令序列进行用户行为模式提取并据此对用户行为进行识别分类。

然而由于该方法算法复杂度相对较高，当 Hadoop 平台用户数量较多时，将用户行为模式提取及识别过程全部放在主节点上进行会使主节点资源消耗过量，容易导致主节点负载过大，从而影响 Hadoop 平台的整体效率。本文利用 Hadoop 自身的 MapReduce 框架对以上算法进行改进，将用户行为模式提取进行并行化处理，结合用户行为评估合理性要求设计了一种适用于 Hadoop 平台的用户行为评估方法，该方法的流程如图 3 所示。

### 3.2 并行指令序列学习

由于本文中的用户行为为指令行为，故用户行为序列 $S$ 是用户指令序列的总称。将收集到的所有用户的指令操作记录（*UserId*，*Cmd*，*Time*）滤除指令参数等信息，仅保留指令名称后，使每个用户固定数量的指令名称按照时间信息排成一个指令流，称为用户指令序列块 $B$，某用户所有的指令序列块首尾连接构成该用户的用户行为序列 $S$（如序列 *cppshxrdbcpp⋯unameacroread*）。

1) 串表压缩算法和字典压缩算法[7]

串表压缩算法（LZW, Lempel-Ziv-Welch）通过建立一个字符串编码字典，用较短的编码代替较长的字符串实现压缩。通过该算法中的生成字符串编码字典的方法，从用户指令序列块 $B$ 中提取序列中

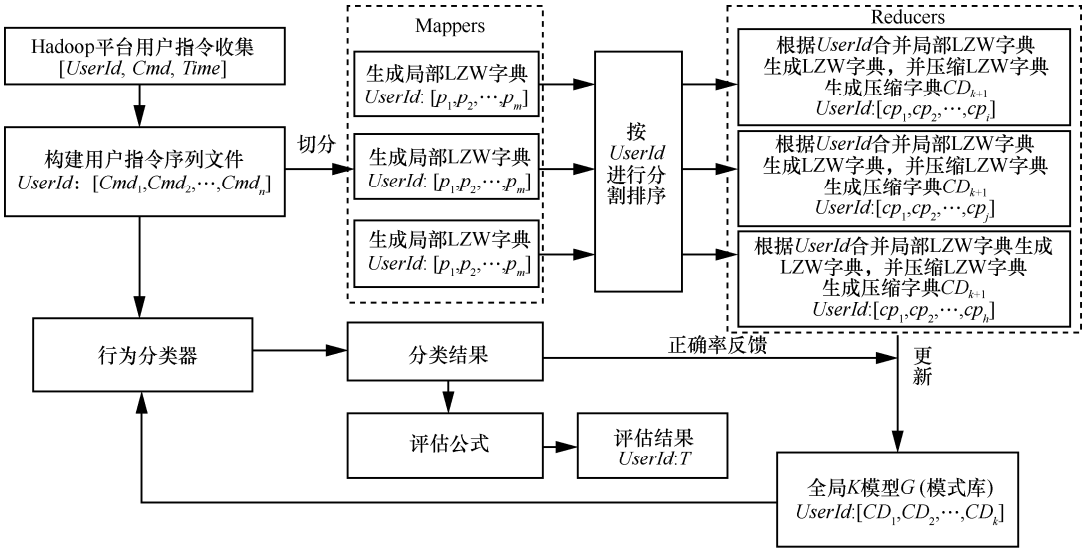


图 3 Hadoop 平台的用户行为评估

所有可能的序列组合即用户指令序列模式 $p_i$，计算模式 $p_i$ 在当前序列块中出现的次数 $f_i$，得到 LZW 字典 $D\{p_i, f_i\}$。

字典压缩算法按照模式在其指令块 $B$ 中的权重 $\omega_i$ 和模式长度 $L_i$ 提取最终的用户行为模式 $cp$，组成压缩字典（CD, compressed dictionary），其中，权重 $\omega_i$ 为

$$\omega_i = \frac{f_i}{\sum_{i=1}^{n} f_i} \tag{1}$$

其中，$\omega_i$ 是模式 $p_i$ 在当前序列块中所占的权重，$f_i$ 是模式 $p_i$ 在当前序列块中出现的次数，$n$ 是当前块中互不相同的模式的数量。

2) MapReduce 并行化

MapReduce 是 Hadoop 云平台的核心计算模型[1]，它将运行于大规模集群上的复杂的并行计算过程高度抽象为 2 个函数：Map 函数和 Reduce 函数。一个 Map/Reduce 作业会把输入数据集切分为若干个独立的数据块，由 Map 任务以完全并行的方式对它们进行处理。框架会先对 Map 任务的输出进行排序，然后把结果输入到 Reduce 任务；同时，整个框架负责任务的监控和调度。

在本文研究中，Map/Reduce 框架由一个单独 JobTracker（位于主节点）和若干 TaskTracker（位于各个从节点）共同组成。主节点负责调度整个模式提取任务，而各个从节点仅负责执行由主节点指派的任务。为了尽量减少 MapReduce 任务额外增加的磁盘读取及网络传输消耗，本文设计采用单 MapReduce 任务对整个用户行为模式提取过程进行并行化改进，改进后的并行化指令序列算法如算法 1 所示。

**算法 1**　并行化指令序列学习算法

**输入**　file (*UserId*: $Cmd_1$, $Cmd_2$, … , $Cmd_n$)

**输出**　file(*UserId*: $cp_1$, $cp_2$, … , $cp_h$)

```
map (UserId,cmdstring);
start←0, end ←1;
p=(cmdstring[start]…cmdstring[end]);
if p∈PD then
        build (UserId, p);
        end ← end +1;
else
        PD ← p;
        build (UserId,p);
        start ← end;
        end ← end +1;
end if
reduce(UserId,(p1,p2,…,pm));
D ← 0;
for each p∈(p1,p2,…,pm) do
        if p ∈ D then
                f ← getfreq(D(pi))+1;
                D ← (pi,f);
        else
                D ← (pi, 1);
        end if
end for
CD ← compressdic(D);
for each UserId do
        build(UserId,(cp1,cp2,…,cph));
end for
```

其中，map 函数负责分割序列的局部字典构建，而 reduce 函数负责全局字典的构建和最终压缩字典的生成；build 子函数用于构建键值对，getfreq 子函数用于获取字典 $D$ 中模式 $p$ 的频数，compressdic 子函数即为压缩字典生成函数。Mapper 和 Reducer 的数量是 MapReduce 框架性能的重要制约因素，最佳数量可以通过实验分析后确定。

### 3.3　分类器

对于用户行为序列 $S$ 和全局模型 $G=\{CD_1, CD_2, \cdots, CD_k\}$，若序列 $S$ 中的模式 $sp$ 与 $CD_i$ 中的任意模式 $cp$ 的编辑距离均大于 $x$·lengthof（$cp$）（其中，$x$> 0.3，lengthof 函数用于计算模式的字符长度），则 $CD_i$ 判定其为异常。$K$ 个 $CD$ 通过投票判定最终模式 $sp$ 是否异常，若 $K$ 为偶数且 $CD$ 中判定异常的模型数为 $\frac{K}{2}$，则根据该模式所在集群主机的前向模式是否异常进行判定（前向模式异常则为异常，否则为正常），剩余模式均为正常模式。

将序列块 $B$ 中所有分类后的用户行为模式标定正常（1）、异常（0），根据模式中首条指令的收集时间进行排序存入该用户行为库中，待后续进行用户行为评估。

### 3.4　全局 *K* 模型更新

Hadoop 用户的行为习惯是随时间变化的，如果全局 $K$ 模型不随时间更改将会增加将用户正常行为序列误判为异常行为序列的概率，称为概念漂移[8]。

为了减少概念漂移的影响，所设置的 $K$ 个全局模型应随着用户新指令序列块的加入不断更新。替换规则为新加入的 $K$+1 模型替换掉在最近的判定中出现错误判定次数最多的模型。

### 3.5 综合评估公式

得到带有分类标签的用户行为模式序列后，需要依次对其进行用户综合评估值的计算，本文设计用户综合行为评估值计算式为

$$T = T_s + \alpha V_n + \beta V_p + \gamma V_r \tag{2}$$

其中，$T$ 为用户综合评估值，$T_s$ 为 Hadoop 平台给所有用户设置的初始评估值，$V_n$ 为当前行为的行为评估值，$V_p$ 为历史行为评估值，$V_r$ 为推荐行为评估值；$\alpha$、$\beta$ 和 $\gamma$ 分别为 $V_n$、$V_p$ 和 $V_r$ 的权重，按照用户行为评估原则，三者应满足 $\alpha>\beta>\gamma$ 且 $\alpha+\beta+\gamma=1$。

$V_n$ 的计算式为

$$V_n = w + \lambda(-\theta j) \tag{3}$$

其中，$w$ 为常数；$0 \leqslant \theta \leqslant 1$，$\theta$ 用于调节异常行为对行为评估值 $V_n$ 的影响作用大小；$j$ 是该用户有效记录中进行异常操作的重复次数；$\lambda$ 为选择因子，当前行为为异常行为时，$\lambda$=1；否则，$\lambda$=0。

$V_p$ 的计算采用滑窗算法[9]，其中，滑窗左沿以外的用户行为记录为过期记录，滑窗右沿设置到当前用户行为的左侧。评估值计算时仅计算滑窗内的用户行为的评估值，计算式为

$$V_p = \sum_{i=1}^{L}\left[\frac{i+1}{\sum_{i=0}^{L}(i+1)}\right]V \tag{4}$$

其中，$L$ 为滑窗长度，基本思路是越近期的行为，其在综合评估中所占权重越大。

$V_r$ 的计算如式(5)所示，取所有推荐值的平均值。

$$V_r = \frac{\sum_{i=1}^{k}[V_{ri}]}{k} \tag{5}$$

计算结束后，将综合评估值存入数据库中相应用户的综合评估值字段。

## 4 实验与评估

为了验证 DACUBA 模型中用户行为评估方法及访问控制策略的有效性、模型本身的可行性和模型对云平台的性能影响，本文设计并进行了实验验证。

### 4.1 数据集

将 Schonlau 数据集中部分指令序列固定替换为 Hadoop 平台云服务指令序列获得实验数据集。该数据集涉及了 50 个用户，每个用户包含 15 000 条指令。

每个用户的前 5 000 条指令中不包含异常指令序列。本文研究中对异常指令的定义为：针对某一特定用户的指令，不符合某一用户正常行为轮廓的指令序列，对该用户来说即为异常序列。例如，从用户 1 中截取一段指令序列 $A$ 插入用户 2 指令序列中，则序列 $A$ 对用户 2 来说就是异常指令序列；本实验数据集正是基于此理论构建，类似的数据集还有 AT& T Shannon 实验室 SEA 数据集。

在每个用户的 15 000 条指令中，后 10 000 条指令掺杂了不同数量的异常指令序列。在数据预处理过程中，将每个用户的所有指令序列划分为 150 个指令序列块 $B$，每个序列块包含 100 条指令，前 50 个指令序列块用于训练，后 100 个序列块用于检测分类测试。

### 4.2 平台实现方案

在 Hadoop1.2.1 版本源码基础上，实现了 DACUBA 模型。具体技术实现方案如下。

1) 在主节点使用 XML 建立简易用户数据库，用于存储用户行为模式序列、行为评估值和综合评估值，并实现简易数据库的增、删、改、查接口。

2) 在主节点安装 Linux 开源 acct 工具对平台内所有用户的指令序列(*UserId*, *Cmd*, *Time*)进行收集存入 log 文件，编写 Python 脚本对 log 文件进行解析后以(*UserId*:*cmdstring*)格式存入行为记录文件。

3) 在主节点按照 MapReduce 框架实现并行化用户行为序列分类算法，并将分类结果存入行为模式库中。

4) 在主节点添加 behavior_assess 类，读取分类结果，实现用户行为评估，并将行为评估值存入评估值数据库。当综合行为评估值小于阈值时，向 hadoop-root-namenode-master.log 日志中写入记录。根据综合行为评估值与阈值的关系并依据 DACUBA 模型中的动态访问控制策略进行相应的操作。

将修改后的 Hadoop 代码重新编译后在实验室平台上进行部署，平台由 3 台物理主机组成，其中一台物理主机作为主节点，另外 2 台物理主机分别安装 2 台虚拟机共 4 台虚拟机作为从节点，配置如下。

物理主机配置：CPU 2.4 GHz，4 核，12 GB 内存，1 TB 磁盘空间，64 bit 总线。

虚拟机配置：CPU 2.4 GHz，双核，1 GB 内存，200 GB 磁盘空间，64 bit 总线。

操作系统：Ubutun12.04L-x64。

### 4.3　参数设置

PCSL 算法中，Mapper 和 Reducer 的数量是用户行为模式提取效率的重要影响因素，为了得到较为合理的 Mapper 和 Reducer 数量（*numM* 和 *numR*），本文设计了实验对两者不同的取值情况下用户行为分类时间进行了统计。得出的实验结果如图 4 所示。

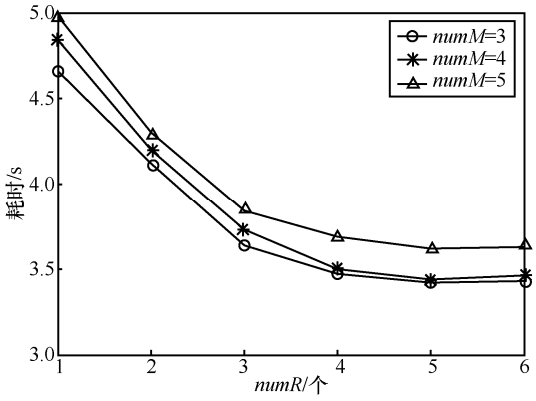


图 4　Mapper 和 Reducer 数量与用户行为模式提取效率的关系

由图 4 可知，在 *numM* 相同的条件下，随着 *numR* 数量的递增，用户行为模式提取效率递增，耗时减少，但当达到一定数量后耗时不再减少，反而会出现一定程度的增加。当数据量较大时，Mapper 的数量应与从节点数量相当。综合以上两点，在本研究中选取 *numM* 为 4，*numR* 为 5。

在 DACUBA 模型中，全局 $K$ 模型中 $K$ 值的设定同样由实验获得，综合实验结果和时间复杂度分析，将 $K$ 值设置为 5。实验中的其他参数取值如表 1 所示。其中，行为检测阈值 $x$ 的选取需平衡异常行为检测率和误报率，使两者均处于可接受范围内；$\alpha$、$\beta$ 和 $\gamma$ 的选取符合当前行为、历史行为及推荐评估值对综合行为评估值的影响程度；其他同评估过程相关的参数可适当放大缩小，但应满足以下两点要求。

1) 所选参数应能体现第 2 节所述用户行为评估的合理性要求。

2) 正常用户在滑窗长度 $L$ 内被误判为异常指令序列模式的平均个数引起综合行为评估值下降后，其综合评估值仍在阈值 $T_d$ 之上，增加冗余性。

**表 1**　　**参数设置**

| 参数 | 取值 |
|---|---|
| 行为检测阈值 $x$ | 0.4 |
| 行为评估基准值 $w$ | 0.5 |
| 异常行为调节参数 $\theta$ | 0.4 |
| 初始行为评估值 $T_s$ | 5 |
| 滑窗长度 $L$ | 100 |
| 推荐评估值 $V_r$ | 2 |
| 当前行为权重 $\alpha$ | 0.6 |
| 历史行为权重 $\beta$ | 0.3 |
| 推荐评估值权重 $\gamma$ | 0.1 |
| 阈值 $T_d$ | 4.5 |

### 4.4　行为评估结果

1) 行为分类实验

首先，采用 Java 编程分别实现本文 PCSL 行为序列分类算法、朴素贝叶斯分类算法[10]、马尔可夫链模型分类算法[11]和共生矩阵分类算法[12]。然后，在实验中输入数据集中每个用户的前 50 个指令序列块进行训练，对后 100 个序列块进行分类，运行 PCSL 行为序列分类算法程序，统计对用户行为的异常行为检测率、异常行为误报率及运行时间。最后，采用相同方法获取朴素贝叶斯分类算法、马尔可夫链模型分类算法和共生矩阵分类算法的分类结果。上述 4 种方法的分类结果如表 2 所示。

**表 2**　　**分类结果对比**

| 分类方法 | 异常行为检测率 | 异常行为误报率 | 时间/s |
|---|---|---|---|
| 朴素贝叶斯 | 84.46% | 7.12% | 49.38 |
| 马尔可夫链模型 | 88.24% | 1.24% | 91.85 |
| 共生矩阵 | 90.02% | 1.16% | 201.23 |
| PCSL | 87.59% | 1.33% | 3.42 |

由表 2 可见，PCSL 算法对用户异常行为的检测率高于朴素贝叶斯算法，略低于马尔可夫链模型算法和共生矩阵算法；在误报率方面，PCSL 算法远低于朴素贝叶斯算法，略高于马尔可夫链模型算法和共生矩阵算法。但是，PCSL 算法在时间开销上远远小于其他 3 种算法，说明该算法具有较大的时间性能优势。针对本文实时监控的应用需求，在对异常行为检测率和误报率与马尔可夫链模型算法和共生矩阵算法相差不大的前提下，PCSL 算法耗时较少，可较好满足实时应用需求。

2) 行为评估算法实验

首先，使用 DACUBA 模型的评估算法对实验

中所获得的正常用户（用户 1）与异常用户（用户 3）的行为分类结果进行行为评估，得到 2 个用户的综合评估值。然后，将 2 个用户前 200 个分类后的测试指令序列模式对应的用户综合评估值数据导入 Matlab，在 Matlab 中处理并得到 2 个用户的综合行为评估曲线，如图 5 所示。

从图 5 中可知，当用户出现异常行为时，其综合行为评估值快速下降，而正常用户行为不会因为累积而大量提高用户综合行为评估值；正常用户综合行为评估值在小范围内波动，而异常用户综合行为评估值波动范围较大。

由此可知，本文的用户行为检测与评估算法能够与 DACUBA 模型有效结合，对 Hadoop 平台中的用户行为做出较为合理的评估。

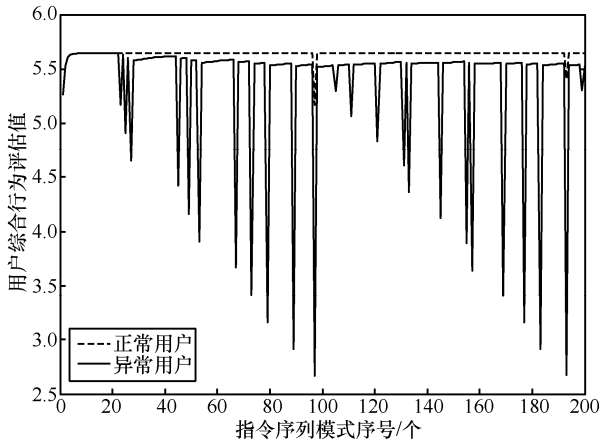


图 5 用户综合行为评估值曲线

### 4.5 有效性验证实验及性能开销

1) 有效性验证

有效性验证实验的实验过程如下。

**步骤 1** 将用户 1 和用户 3 的前 50 个序列块中的指令依次执行进行训练。

**步骤 2** 通过配置开启平台用户行为评估模块和动态访问控制策略模块。

**步骤 3** 用户 1 继续输入数据集中部分后续指令，用户 3 执行掺杂有异常指令序列的指令块后，两者对 Hadoop 文件系统中测试文件进行 cat 读操作，用户 1 可以正常读取，而用户 3 给出不能访问提示。

**步骤 4** 查看主节点的 log 文件，发现用户 3 的综合评估值过低的警告记录。

**步骤 5** 进行 MapReduce 操作测试，结果与步骤 4 类似。

有效性验证实验结果如图 6 所示。图 6 的结果证明本文方法能够有效动态控制平台用户的访问请求。

```
user3@node1:/home/hadoop-1.2# ./bin/hadoop fs -cat ./test_in/test.txt
hadoop:Your comprehensive evaluation value is too low,Please contact the administrator
user3@node1:/home/hadoop-1.2#
user3@node1:/home/hadoop-1.2# ./bin/hadoop jar hadoop-examples-1.2.1.jar wordcount test_in test_out
hadoop:Your comprehensive evaluation value is too low,Please contact the administrator
user3@node1:/home/hadoop-1.2#
```

(a) 命令行禁止访问提示

```
2016-05-29 09:41:00,508 INFO org.apache.hadoop.hdfs.StateChange: Removing lease on  /opt/had
oop-1.2/mapred/system/jobtracker.info from client DFSClient_NONMAPREDUCE_-2033573953_1
2016-05-29 09:41:00,508 INFO org.apache.hadoop.hdfs.StateChange: DIR* completeFile: /opt/had
oop-1.2/mapred/system/jobtracker.info is closed by DFSClient_NONMAPREDUCE_-2033573953_1
2016-05-29 09:41:02,854 INFO org.apache.hadoop.hdfs.StateChange: BLOCK* ask 10.1.80.242:5001
0 to delete  blk_1226714346986192493_1090
2016-05-29 09:41:05,855 INFO org.apache.hadoop.hdfs.StateChange: BLOCK* ask 10.1.80.240:5001
0 to delete  blk_1226714346986192493_1090
2016-05-29 09:41:05,857 INFO org.apache.hadoop.hdfs.StateChange: BLOCK* ask 10.1.80.241:5001
0 to delete  blk_1226714346986192493_1090
2016-05-29 09:50:34,657 INFO org.apache.hadoop.accessmanger: The user(uid:1006) Comprehensiv
e evaluation value is too low!!! T=4.350
```

(b) log 文件警告记录

图 6 有效性验证实验结果

2) 性能开销验证

在实验中，为了模拟大量用户的同时访问 Hadoop 平台，将每个用户后 100 个序列块进行重复拷贝，编写脚本模拟多个用户对平台进行命令输入，总输入速度为 0.1 MB/s（约为 1 000 个用户同时访问 Hadoop 平台产生的命令序列字节大小），测试并记录部署 DACUBA 模型的云平台对文本文件的 HDFS 读、写操作和 MapReduce WordCount 操作的效率。然后，在相同环境下，在原 Hadoop 平台上执行相同的测试操作。2 个平台的操作耗时对比结果如表 3 所示。

**表 3 耗时情况对比**

| 平台 | 操作 | 速率/(MB·s$^{-1}$) |
|---|---|---|
| 原 Hadoop 平台 | HDFS 读 | 41.426 |
| | HDFS 写 | 20.562 |
| | MapReduce WordCount | 0.415 |
| 部署 DACUBA 的 Hadoop 平台 | HDFS 读 | 39.859 |
| | HDFS 写 | 19.632 |
| | MapReduce WordCount | 0.393 |

由表 3 可知，与原 Hadoop 平台相比，DACUBA 模型对 HDFS 读、写操作及 MapReduce 操作的速度影响较小，且均在可接受范围之内。

## 5 结束语

针对 Hadoop 云平台用户的安全威胁问题，本文提出一种基于用户行为评估的动态访问控制模型（DACUBA）。该模型通过检测用户指令行为序

列，对用户进行实时的行为监控和评估，从而动态地控制用户对 Hadoop 平台的云服务请求。实验结果证明，该模型的行为分类方法准确率较高且评估方法较为合理，时间和性能开销较小，该模型访问控制策略能够有效地与行为评估算法结合实现对云平台用户的动态访问控制。

## 参考文献：


[1] 陆嘉恒. Hadoop 实战[M]. 北京：机械工业出版社, 2012.
LU J H. Hadoop in action[M]. Beijing: China Machine Press, 2012.

[2] GUPTA C, SINHA R, ZHANG Y. Eagle: user profile-based anomaly detection for securing Hadoop clusters[C]//IEEE International Conference on Big Data. 2015:1336-1343.

[3] TAN Z, TANG Z, LI R, et al. Research on trust-based access control model in cloud computing[C]//Information Technology and Artificial Intelligence Conference. 2011:339-344.

[4] JING X, LIU Z, LI S, et al. A cloud-user behavior assessment based dynamic access control model[J]. International Journal of System Assurance Engineering & Management, 2015, 22(12): 1-10.

[5] ZBURIVSKY D. Hadoop 集群与安全[M]. 北京：机械工业出版社, 2014.
ZBURIVSKY D. Hadoop cluster deployment, securing Hadoop[M]. Beiing: China Machine Press, 2014.

[6] JAIGANESH M, AARTHI M, KUMAR A V A. Fuzzy ART-based user behavior trust in cloud computing[J]. Advances in Intelligent Systems & Computing, 2015, 324:341-348.

[7] CHUA S L, MARSLAND S, GUESGEN H W. Unsupervised learning of patterns in data streams using compression and edit distance[C]//The International Joint Conference on Artificial Intelligence. 2011:1231-1236.

[8] KRANEN P, KREMER H, JANSEN T, et al. Stream data mining using the MOA framework[C]//International Conference on Database Systems for Advanced Applications. 2012:309-313.

[9] 王昕, 袁超伟, 黄晨. 基于滑动窗口机制的 RFID 自同步可扩展所有权变更协议[J]. 北京邮电大学学报, 2013, 36(5):30-35.
WANG X,YUAN C W, HUANG C. Scalable and self-synchronizable RFID ownership transfer protocol based on the sliding window mechanism[J]. Journal of Beijing University of Posts and Telecommunications, 2013, 36(5):30-35.

[10] MAXION R A, TOWNSEND T N. Masquerade detection using truncated command lines[C]//International Conference on Dependable Systems and Networks. 2002:219-228.

[11] TIAN X G, DUAN M Y, LI W F, et al. Anomaly detection of user behavior based on shell commands and homogeneous Markov chains[J]. Chinese Journal of Electronics, 2007, 17(2):231-236.

[12] 李超, 田新广, 肖喜,等. 基于 Shell 命令和共生矩阵的用户行为异常检测方法[J]. 计算机研究与发展, 2012, 49(9):1982-1990.
LI C, TIAN X G, XIAO X, et al. Anomaly detection of user behavior based on shell commands and commands and co-occurrence matrix[J]. Journal of Computer Research and Development, 2012, 49(9): 1982-1990.


## 作者简介：

**杨宏宇**（1969-），男，吉林长春人，博士，中国民航大学教授，主要研究方向为网络信息安全。

**孟令现**（1990-），男，山东临沂人，中国民航大学硕士生，主要研究方向为云平台安全。