

大数据环境下多源异构数据的访问控制模型

苏秋月¹, 陈兴蜀², 罗永刚³

- (1. 四川大学计算机学院, 四川 成都 610065;
2. 四川大学网络空间安全学院, 四川 成都 610065;
3. 四川大学网络空间安全研究院, 四川 成都 610065)

摘要: 大数据平台具有开放性和共享性, 但随着数据量不断增加且用户访问上下文环境复杂多变, RBAC模型难以满足大数据环境下细粒度、灵活的访问控制。针对这一问题, 提出了大数据平台下多源异构数据的访问控制模型。该模型根据属性动态地确定角色权限, 并构建基于数据组的层次结构, 实现数据属性的简单管理。对该模型进行了形式化定义, 阐述了在 Hadoop 平台中的实现和 workflows, 并通过实验验证了所提方案的性能开销相对较小。

关键词: 大数据平台; 访问控制; 动态授权; 数据组层次结构

中图分类号: TP309.2

文献标识码: A

doi: 10.11959/j.issn.2096-109x.2019009

Access control model for multi-source heterogeneous data in big data environment

SU Qiuyue¹, CHEN Xingshu², LUO Yonggang³

1. College of Computer Science, Sichuan University, Chengdu 610065, China
2. College of Cybersecurity, Sichuan University, Chengdu 610065, China
3. Cybersecurity Research Institute, Sichuan University, Chengdu 610065, China

Abstract: The big data platform is open and shared, but with the increasing amount of data and the complex and variable user access context, the RBAC model is difficult to meet the fine-grained and flexible access control in big data environment. To solve this problem, an access control model for multi-source heterogeneous data in big data platform is proposed. The model dynamically determines role permissions based on attributes and builds a hierar-

收稿日期: 2018-11-28; **修回日期:** 2018-12-17

通信作者: 陈兴蜀, chenxsh@scu.edu.cn

基金项目: 国家自然科学基金青年科学基金资助项目 (No.61802270, No.61802271); 国家“双创”示范基地之变革性技术国际研发转化平台资助项目 (No.C700011); 中央高校基本科研业务费基础研究项目 (No.2017SCU11059, No.SCU2018D018)

Foundation Items: The National Natural Science Foundation of China Youth Science Fund Project (No.61802270, No.61802271), National “Double Creation” Demonstration Base's Transforming Technology International R&D and Transformation Platform Funding Project (No.C700011), The Fundamental Research Funds for the Central Universities (No.2017SCU11059, No.SCU2018D018)

论文引用格式: 苏秋月, 陈兴蜀, 罗永刚. 大数据环境下多源异构数据的访问控制模型[J]. 网络与信息安全学报, 2019, 5(1): 78-86.

SU Q Y, CHEN X S, LUO Y G. Access control model for multi-source heterogeneous data in big data environment[J]. Chinese Journal of Network and Information Security, 2019, 5(1): 78-86.

chical structure based on data groups to achieve simple management of data attributes. The model is formally defined, and the implementation and workflow in Hadoop platform are described. The experimental results show that the performance overhead of the proposed scheme is relatively small.

Key words: big data platform, access control, dynamic authorization, data group hierarchy

1 引言

大数据平台不仅是分布式、高度可扩展的平台,还支持多源异构数据源的集成。随着大数据、云计算的不断发展,越来越多的企业将其内部数据存放在公共大数据平台中,以降低公司的运行成本。与此同时,如何保障平台中的数据不被其他租户非法访问,确保数据的安全性成为企业用户关心的问题。因此,如何实现云存储中多租户的数据安全隔离一直是业界研究的重点。

访问控制是实现数据受控访问、保护数据安全的有效手段之一。在存储海量多源异构数据的大数据开放平台上,为了实现租户之间数据的隔离访问,应实施高效、细粒度且灵活的访问控制。但在大数据平台中对多源异构数据的访问控制面临如下问题:首先大数据平台上汇聚了来自不同租户的异构数据,虽然基于Hadoop的大数据平台提供了不同的组件来存储这些多源异构的数据,但大多只能依赖访问控制列表(ACL, access control list)^[1]实现粗粒度的访问控制,在使用过程中权限不易管控,存在用户可以擅自提权的风险;其次平台中用户、数据以及访问上下文环境在时刻变化,访问权限也需要随之变化,否则容易出现授权不足或过度授权的问题。

近年来,国内外企业和学者对Hadoop平台的访问控制问题进行了深入研究。Hortonworks公司和Cloudera公司相继推出了Hadoop平台上集中访问控制的开源框架,分别是Apache Ranger^[2]和Apache Sentry^[3]。其中,Ranger采用

基于策略的访问控制模型,通过设置策略为用户或用户组授予对数据的访问权限。Sentry采用基于角色的访问控制(RBAC, role-based access control)模型^[4]。RBAC模型通过分配给用户的角色来授予用户访问的权限,并且可以通过构建角色层次结构来减少用户-角色分配,实现了对访问控制系统的简单管理。但在多租户环境下,用户-角色关系和角色-权限关系经常变化,安全管理员难以考虑所有的数据访问权限,使权限分配难度增加。

Gupta等^[5-6]总结Ranger、Sentry以及Hadoop原生基于ACL这3种访问控制机制的特点,提出了Hadoop生态系统的多层授权框架和Hadoop生态系统授权模型。陈垚坤等^[7]和Gupta等^[8]将基于属性的访问控制(ABAC, attribute-based access control)^[9]模型应用到Hadoop平台中,根据用户、数据、环境以及操作的属性制定访问控制策略,其中,相应属性可以包含实体的任何描述符,如标签、安全级别等。与RBAC模型相比,ABAC模型利用基于属性的访问控制策略直接定义了用户权限关系,实现了更加灵活的访问控制,在一定程度上避免了RBAC中权限分配困难的问题。但正是由于ABAC模型灵活的授权管理,目前属性的定义没有一个较好的格式标准,导致属性管理难度较大。另外,访问权限判定的时间消耗也取决于属性选择的复杂性。

文献[10-12]将RBAC模型与主客体标签结合,提出了一种面向多租户云存储平台的访问控制策略。该策略通过比较主客体标签是否相符,确保了租户之间数据的强隔离性,但随着用户和数据不断变化,这种强制访问控制机制下的访问

权限难以灵活地随之改变。Bhatt 等^[13]将角色作为属性之一添加到 ABAC 模型中, 提出在多租户云环境下的一种属性角色结合的访问控制机制。王于丁等^[14]进一步提出了一种基于角色和属性的云计算数据访问控制模型。最近, Gupta 等^[15]在 Hadoop 生态系统中讨论了角色与属性结合的方法, 将对象标签作为对象属性添加到 RBAC 模型中, 提出基于对象标签的角色访问控制模型。上述研究将属性与 RBAC 结合使用, 实现了简单的授权管理和动态的权限分配, 但均未解决属性管理复杂的问题。

针对上述问题, 本文提出了大数据平台下多源异构数据的访问控制模型 (MT-MHAC, access control model for multi-source heterogeneous data in big data platform)。该模型通过构建数据组层次结构扩展 RBAC 模型, 避免了冗余的角色权限分配; 采用基于属性的策略定义角色权限关系, 细粒度、灵活地分配角色的权限; 另外, 将相同数据属性的数据分配到同一数据组, 数据可以通过数据组的层次结构继承数据属性, 避免为数据设置重复属性, 简化了数据属性的管理。最后, 本文将 MT-MHAC 模型实现在 Hadoop 平台中。通过实验验证, 该模

型可以满足多租户环境下, 对多源异构数据细粒度、动态授权的访问控制需求, 保障租户之间数据的安全隔离。

2 MT-MHAC 模型

2.1 模型元素

RBAC 模型为每个用户分配一个或多个角色, 并通过定义角色层次结构减少了用户角色的分配工作。随着大数据向更多应用领域扩展, 越来越多的数据涌入大数据平台中, RBAC 模型下的角色-权限分配成为一项重大挑战。本文旨在 Hadoop 平台中多租户场景下, 对多源异构数据实施灵活、细粒度的访问控制。本文使用数据治理工具 Apache Atlas^[2]对平台中的多源异构数据进行集中管理, 并在此前提下提出了 MT-MHAC 模型, 如图 1 所示。

在基于 Hadoop 的大数据平台中, 很多组件中数据具有天然的层次结构。例如, 分布式文件系统 (HDFS) 中的文件存在于目录树下; 数据仓库工具 Hive 中数据有数据库、表、行或列这几种类型, 也存在层次结构。MT-MHAC 模型利用了平台中这些数据的特点, 构建了数据组的层次结构, 与 RBAC 模型的角色层次结构相

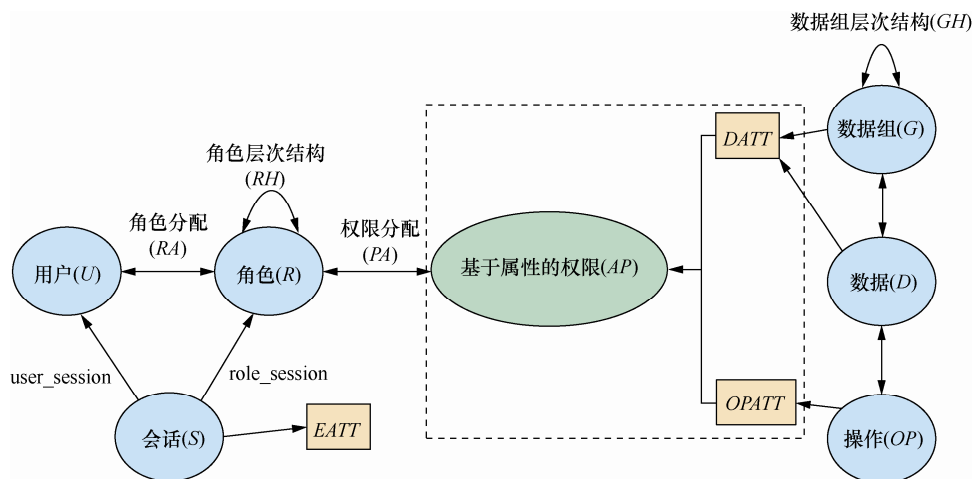


图 1 MT-MHAC 模型

结合,简化了角色-权限的分配,可以将平台中具有相同属性的数据分配到同一数据组中,为该组分配适当的属性和属性值,而不用为每个数据分配属性。如果数据组之间还存在层次结构,可以间接通过其他数据组继承数据属性,通过这种数据组层次结构简化了数据属性的管理。另外,采用基于属性的权限策略,实现对多源异构数据灵活简单的访问控制,确保了租户间数据安全灵活的隔离。

为了保证 MT-MHAC 模型的正常运行,假设访问 Hadoop 平台的用户均已通过身份认证。本文所提出的模型中实体包括用户 (U)、角色 (R)、会话 (S)、数据 (D)、数据组 (G)、操作 (OP)。属性集合包括数据属性 ($DATT$)、环境属性 ($EATT$) 和操作属性 ($OPATT$)。具体定义描述如表 1 所示。

2.2 实体间的关系

除了实体之外,模型中还包含实体间的关系,通过关系集合将各个实体集联系起来,实现实体中元素之间的操作。下面对这些关系进行定义。

定义 1 角色层次结构 $RH \subseteq R \times R$, 表示在角色集合 R 上存在的偏序关系。角色的继承从低到高,即父角色继承其所有的子角色。若存在

$r_i \succ r_j$, 其中 $r_i, r_j \subseteq R$, 则 r_i 是 r_j 的父角色。这里 r_i 不仅具有其自身角色,还继承了 r_j 的角色。

例如,用户 Gary 被分配角色为 *developer*, 用户 Bard 被分配角色为 *direct*, 其中, *direct* 是 *developer* 的父角色。因此用户 Bard 所拥有的角色除了被直接分配的 *direct* 之外,还有通过继承得到的 *developer*。由此可知,角色层次结构可以方便地为单个用户分配和删除多个角色,简化了用户-角色关联关系的管理。

定义 2 数据-数据组分配 $DGA \subseteq D \times G$, 等价于 $directG(d): D \rightarrow 2^G$, 表示数据与数据组之间多对多的映射关系。可以将数据分配给多个数据组,通过函数 *directG* 获取数据并返回该数据所属的数据组集合。

定义 3 数据组层次结构 $GH \subseteq G \times G$, 表示在数据组集合 G 上的偏序关系。若 $g_i \succ g_j$, 其中 $g_i, g_j \subseteq G$, 意味着分配了数据组 g_i 的角色将获得该组和来自低级数据组 g_j 这 2 个组中所有数据的访问权限。

大数据环境下,租户将数据源源不断地从各种系统聚集到 Hadoop 平台中,同时用户访问数据的上下文在不断变化,导致用户的访问权限需要及时更新。但在 RBAC 模型中,用户权限的

表 1 实体和属性定义

名称	定义及描述
U	用户集合: $U=\{U_1, U_2, \dots, U_n\}$, U_i 表示 Hadoop 平台中的某一用户
R	角色集合: $R=\{R_1, R_2, \dots, R_n\}$, R_i 表示 Hadoop 平台中的某一角色
S	会话集合: $S=\{S_1, S_2, \dots, S_n\}$, S_i 表示用户的一次访问请求,与激活角色之间的某种映射
D	数据集合: $D=\{D_1, D_2, \dots, D_n\}$, D_i 表示 Hadoop 平台中的某一数据
G	数据组集合: $G=\{G_1, G_2, \dots, G_n\}$, G_i 表示平台中具有相同属性的某一数据集合
OP	操作集合: $OP=\{OP_1, OP_2, \dots, OP_n\}$, 表示对数据的操作集合
$DATT$	数据属性集合: 对 Hadoop 平台中数据的特征描述,如数据的标识、类型、拥有者等
$EATT$	环境属性集合: 表示 Hadoop 平台中执行访问操作的上下文信息,如时间、位置等
$OPATT$	操作属性集合: 表示用户对多源异构数据执行具体行为的描述,如读/写,创建/删除等

更新只能通过修改或增加为用户所分配的角色，而角色的权限更新需要修改原有的角色权限关系，导致授权管理难度增加。为了解决上述问题，本文定义数据属性、环境属性和操作属性，根据属性策略进行角色权限分配，动态地建立角色与权限之间的关系。

定义 4 属性是对实体特征的具体描述，记为 $ATT ::= (att_name \text{ op } att_value)$ 。其中， att_name 是指属性名称； op 指属性运算符，包括 $<$ 、 $=$ 、 $>$ 、 $\leq$ 、 $\neq$ 、 $\geq$ 等； att_value 指属性的取值，可由有限的原子值或集合表示。例如， $\exists datt \subseteq DATT$ ， $datt := (type = \{file, database, table\})$ 表示数据属性集中有一个属性为 $type$ ，其属性值等于一个包含 $file$ 、 $database$ 和 $table$ 的集合。

定义 5 数据组的有效属性 $effectiveG_{datt}$ ，包含直接分配给用户组的数据属性和来自子数据组所继承的属性。对于 $\forall datt \in DATT$ ， $\forall g \in G$ ，有

$$effectiveG_{datt}(g) = datt(g) \cup \left(\bigcup_{\forall g' \in \{g \mid g' \succ g\}} effectiveG_{datt}(g') \right) \quad (2)$$

定义 6 数据有效属性 $effectiveD_{datt}$ ，由自身具有的属性、所属数据组的所有属性，以及数据组通过继承其他组所得的属性这 3 部分组成。当为数据设置属性时，无须为每个数据都设置属性，只需将相同属性的数据添加到同一数据组，然后设置其最小权限范围的属性及属性值。数据组中的数据通过数据组的层次关系间接获得层次化的属性和属性值。本文通过这种数据组层次结构为安全管理员提供了简单灵活的数据属性管理，简化了角色-权限的分配。对于 $\forall datt \in DATT$ ， $\forall g \in G$ ，有

$$effectiveD_{datt}(d) = datt(d) \cup \left(\bigcup_{\forall g \in directG(d)} effectiveD_{datt}(g) \right) \quad (3)$$

定义 7 基于属性的权限 $AP \subseteq DATT \times EATT \times$

$OPATT$ ，其中， $\forall eatt \in EATT$ ， $\forall opatt \in OPATT$ ，数据属性满足 $\forall datt \in DATT \subseteq effectiveD_{datt}(d)$ 。通过定义数据属性、环境属性和操作属性动态地确定数据的访问权限，实现了灵活的授权管理。

定义 8 模型中的函数

1) 用户-角色分配 $assignRole(u): U \rightarrow 2^R$ ，等价于 $UR \subseteq U \times R$ ，其中， $\exists r \in \{r_i \mid r_i \succ r_j, r_i, r_j \in R\}$ ，表示用户到角色多对多的映射关系。

2) 角色-权限分配 $assignPerm(r): R \rightarrow 2^{AP}$ ，表示角色到权限多对多的映射关系。

3) 用户会话 $user_session(u): U \rightarrow 2^S$ ，表示单个用户到会话集合的映射函数。

4) 角色会话 $role_session(s): S \rightarrow 2^R$ ，表示会话到角色集合的函数，其中，会话所对应的角色应满足 $R \subseteq UR$ 。

5) 角色权限判定规则如下：当用户在会话中所获得的角色 r 满足数据 d 的属性、环境属性 $eatt$ 和操作属性 $opatt$ 时，允许角色访问。

$$\begin{aligned} is_authorized(r: R, d: D, op: OP) \Leftarrow & \\ (\exists r \in assignRole(u)) \wedge & \\ (\exists d \in effectiveD_{datt}(d), datt \in DATT) \wedge & \\ (\exists opatt \in OPATT) \wedge (\exists epatt \in EATT) \wedge & \\ ((datt, eatt, opatt) \in assignPerm(r)) & \end{aligned} \quad (4)$$

3 模型实现

为了将 MT-MHAC 模型应用到 Hadoop 平台中，本文利用基于 Apache Atlas 的元数据管理工具来集中管理汇集到平台中的多源异构数据。另外，在基于 Apache Ranger 的权限管理框架上进行了修改和扩展，以实现本文所提出的 MT-MHAC 模型。Ranger 主要由 2 部分组成：策略服务器和插件。策略服务器即权限管理模块，负责管理策略和从 Unix 或 LDAP 同步的用户/用户组，以及为集成到组件中的插件提供策略信息和用户信息。插件是内嵌在组件中的权限判定引

擎, 通过从策略服务器定时轮询获取更新策略来评估访问请求。本文在权限管理模块中定义了访问控制策略决策点 (PDP, policy decision point)、访问控制策略管理点 (PAP, policy administration point) 和访问控制策略信息点 (PIP, policy information point), 并且修改了 Ranger 的权限判定引擎和策略的定义。在权限管理模块中添加用户-角色表、数据属性表和环境属性表, 以便安全管理员定义用户-角色关系、数据属性和环境属性。具体的实现细节如图 2 所示。

在模型的运行过程中, 系统对用户访问数据的控制有 2 个重要阶段。一是授权阶段, 当用户进入 Hadoop 平台时, 安全管理员在权限管理模块中将用户-角色表定义为用户所分配的角色; 然后选择合适数据属性 *datt*、环境属性 *eatt* (如时间、位置等) 和操作属性 *opatt*, 为角色设置基于属性的权限策略, 建立角色与权限之间的关联关系, 这样使用户拥有的角色在不同状态下的权限也不同, 从而可以动态地确定用户的访问权限。二是鉴权阶段, 权限管理模块中的 PIP 先从组件 (图 2 中的 Hive) 中获取当前会话的用户信息和环境信息, 从而根据用户-角色表得到其拥有的角色, 然后通过查询数据属性表、环境属性表和操作属

性表分别得到访问请求中数据的有效属性、环境属性和操作属性, 将其与从 PAP 获取的访问权限策略中对应的策略项进行一一匹配。若均匹配, 则权限管理模块中的 PDP 将权限判定结果 *true* 返回给组件中的 Ranger 插件, 即访问控制策略执行点 (PEP, policy enforcement point), 否则返回 *false*。组件中的 PEP 根据 PDP 返回的结果执行具体的操作。

例如, 用户 *u* 提交访问 Hive 中数据 *d* 的请求, 权限管理模块首先获取用户的访问请求信息 *request(u, d, op)*, 根据环境属性表和操作属性表得到访问请求的环境属性 *eatt* 和操作属性 *opatt*, 再从元数据管理模块中获取 *d* 的属性信息, 然后获取用户-角色表和角色-权限策略。根据访问权限判定算法 (算法 1) 对访问请求 *request* 进行判定, 并将访问判定结果返回给 Hive 中内嵌的 Ranger 插件, 最终由 Ranger 插件执行允许访问或拒绝访问。

算法 1 访问权限判定算法

输入 *u, d, op*

输出 *result*, *result* ∈ boolean

1) *result* = false;

2) *s* = *user_session(u)*;

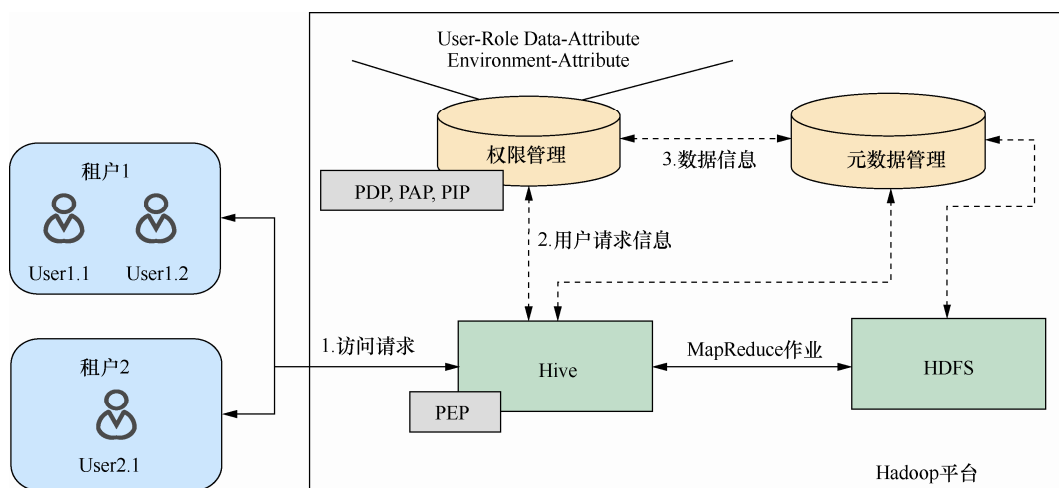


图 2 MT-MHAC 模型在 Hadoop 中的实现框架

```

3)  $r = \text{assignRole}(u)$ ;
4) if  $r = \text{null}$  return result;
5) if  $r \models \text{role\_session}(s) \subseteq r$ 
6)   for each  $ap \in \text{assignPerm}(r')$  do
7)      $datt = \text{effectiveD}_{datt}(d)$ 
8)     if  $(datt, eatt, opatt) \subseteq ap$ 
9)       result = true;
10)      break;
11) return result

```

4 实验分析

4.1 用例

本节以下述应用场景为例说明 MT-MHAC 模型的有效性和优势。现有 2 个公司 TOK 和 SUN 租用同一大数据平台存储企业的数据，这里，同一大数据平台中角色定义和数据属性、环境属性等定义都相同。例如，定义了角色 *developer* 和 *direct*，数据组 *analysis*、*collect* 和 *log*，数据属性 $\text{type} = \{\text{general}、\text{develop}、\text{network}\}$ ，环境属性 $\text{location} = \{\text{china}、\text{sichuan}、\text{beijing}\}$ 。其中， $\text{direct} \succ \text{developer}$ ， $\text{analysis} \succ \text{collect}$ ， $\text{analysis} \succ \text{log}$ 。另外，数据组 *analysis* 包含的属性为 $\text{type} = \{\text{develop}\}$ ，数据组 *collect* 包含的属性为 $\text{type} = \{\text{general}\}$ ，数据组 *log* 包含的属性为 $\text{type} = \{\text{network}\}$ 。根据各公司的访问需求定义了部分权限策略如下。另外，对不同组件中数据的操作不同，如 HDFS 中文件操作为 *r/w/x*，Hive 中数据操作为 *create/delete/update* 等。为了简化权限策略，这里没有定义数据的具体操作，均为 *access*。

Policy1: $\{\text{developer}, \text{type} = \text{network}, \text{location} = \text{sichuan}, \text{op} = \text{access}\}$ 。

Policy2: $\{\text{developer}, \text{type} = \text{general}, \text{location} = \text{beijing}, \text{op} = \text{access}\}$ 。

Policy3: $\{\text{direct}, \text{type} = \text{develop}, \text{location} = \text{beijing}, \text{op} = \text{access}\}$ 。

假设有 TOK 公司员工 Ablett 和 SUN 公司员

工 Carl，且 2 个公司分别在四川和北京。TOK 公司和 SUN 公司的安全管理员分别为 Ablett 和 Carl 分配了角色 *developer* 和 *direct*。通过测试结果表明，根据 Policy1 定义的权限策略，Ablett 只能访问本公司数据组为 *log* 中的数据。由于存在角色继承和数据组继承关系，根据 Policy2 和 Policy3 定义的策略可知，员工 Carl 除了能访问数据组为 *analysis* 的数据之外，还具有数据组 *collect* 中数据的访问权限。通过该角色层次结构和数据组的层次结构，简化了角色分配和数据属性分配，从而为管理员提供了权限的简单管理。另外，虽然 SUN 公司的员工 Carl 通过角色继承拥有了 *developer* 角色，但根据策略中环境属性的定义，其不能访问 TOK 公司数据组为 *log* 中的数据。

上述实验表明，本文所提出的模型根据用户的角色以及角色是否拥有当前会话上下文环境下对数据的访问权限，严格限制租户之间数据的访问，实现了对多租户之间数据的安全隔离；并且通过层次化的角色和数据属性，简化了管理员的权限分配工作；基于属性的权限策略也为租户间及内部提供了细粒度、灵活的访问权限管理。另外，当平台中增加用户和数据或用户权限发生变动时，不用添加角色，通过更新基于属性的权限策略即可实现用户权限的变更。

4.2 性能分析

为了验证 MT-MHAC 模型的性能，本文在 Hadoop 平台中进行了实验测试，其中 Hadoop 版本是 2.7.3，Apache Ranger 版本是 0.7.0。本文通过测试时间开销来分析模型的性能，将 MT-MHAC 模型与 RBAC 模型、ABAC 模型的访问控制决策时间进行对比，分析 MT-MHAC 模型的性能优势。根据访问控制策略判定用户的访问请求是访问控制过程中关键步骤，也是影响访问控制决策时间的主要因素。因此，本文为验证模

型对 Hadoop 平台的性能损失, 将统计不同数量的访问控制策略下 MT-MHAC 模型和 RBAC 模型、ABAC 模型的访问控制决策时间对比, 结果如图 3 所示。

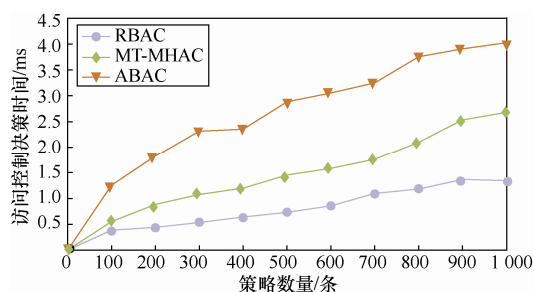


图3 策略数量与访问决策时间对比

随着访问控制策略的增加, 三者的访问控制时间均呈上升的趋势。其中, MT-MHAC 模型的整体访问决策时间与 RBAC 模型相比稍高, 是由于使用数据属性、环境属性和操作来确定角色的权限, 但同时相比 RBAC 模型增加了角色权限匹配的灵活性。另外, 通过添加数据组层次结构简化了角色权限匹配, MT-MHAC 模型的访问决策时间低于 ABAC 模型。从图 3 中可看出, 策略数量 1 000 条以内访问控制决策所用的时间仅在 4 ms 以内。即使策略数量达到上万条甚至几十万条, 访问控制决策的时间开销依然能保持在毫秒级别。由此可知, MT-MHAC 模型的访问控制时间开销对整个访问过程的性能影响较小。

5 结束语

本文针对大数据平台下如何实现多源异构数据的访问控制, 确保租户数据的安全隔离问题, 提出了 MT-MHAC 模型。该模型结合 RBAC 模型中角色层次结构, 增加了数据组的层次结构, 父数据组的权限可以直接从子数据组继承, 简化了角色权限分配; 使用基于属性的策略定义角色与权限的分配关系, 提供了细粒度、灵

活地授权; 可通过数据组层次关系获得其他数据组的继承属性, 提供了数据属性的简单管理。另外, 本文利用 Hadoop 平台中的数据治理工具 Apache Atlas 集中管理这些异构数据, 并在 Apache Ranger 的基础上进行修改, 实现了本文所提出的模型; 并且通过实验性能分析验证了 MT-MHAC 模型在对性能损失基本可以忽略的情况下, 实现了大数据平台中多源异构数据灵活、细粒度的访问控制, 保证了多租户数据之间的安全隔离。

参考文献:

- [1] WHITE T. Hadoop-the definitive guide 4e[M]//Hadoop: The Definitive Guide. O'Reilly Media, Inc. 2015: 1-4.
- [2] 刘驰, 胡柏青, 谢一. 大数据治理与安全: 从理论到开源实践[M]. 北京: 机械工业出版社, 2017.
LIU C, ZHOU B Q, XIE Y. Big data governance and security: from theory to open source practice[M]. Beijing: China Machine Press, 2017.
- [3] SPIVEY B, ECHEVERRIA J. Hadoop security: protecting your big data platform[M]. O'Reilly Media, Inc., 2015.
- [4] SANDHU R S, COYNE E J, FEINSTEIN H L, et al. Role-based access control models[J]. IEEE Computer, 2002, 29(2):38-47.
- [5] GUPTA M, PATWA F, BENSON J, et al. Multi-layer authorization framework for a representative Hadoop ecosystem deployment[C]//ACM on Symposium on Access Control MODELS and Technologies. 2017:183-190.
- [6] GUPTA M, PATWA F, SANDHU R. POSTER: access control model for the Hadoop ecosystem[C]//ACM on Symposium on Access Control MODELS and Technologies. ACM, 2017:125-127.
- [7] 陈垚坤, 刘文丽. 一种适用于 Hadoop 平台的基于属性访问控制模型[J]. 河南师范大学学报(自然版), 2016(5):146-153.
CHEN Y K, LIU W L. Attribute-based access control model for Hadoop[J]. Journal of Henan Normal University (Natural Edition), 2016(5): 146-153.
- [8] GUPTA M, PATWA F, SANDHU R. An attribute-based access control model for secure big data processing in Hadoop ecosystem[C]//ACM Workshop. 2018:13-24.
- [9] HU V, FERRAILOLO D, KUHN R, et al. Guide to attribute based access control (ABAC) definition and considerations[J]. NIST Special Publication, 2013, 800(162).
- [10] 沈晴霓, 杨雅辉, 禹熹, 等. 一种面向多租户云存储平台的访问

控制策略[J]. 小型微型计算机系统, 2011, 32(11):2223-2229.

SHEN Q N, YANG Y H, YU X, et al. An access control policy for multi-tenancy cloud storage platform[J]. Small Microcomputer System, 2011, 32(11): 2223-2229.

- [11] LO N W, YANG T C, GUO M H. An attribute-role based access control mechanism for multi-tenancy cloud environment[J]. Wireless Personal Communications, 2015, 84(3):2119-2134.

- [12] NGO C, DEMCHENKO Y, LAAT C D. Multi-tenant attribute-based access control for cloud infrastructure services[J]. Journal of Information Security & Applications, 2016, 27-28(C):65-84.

- [13] BHATT S, PATWA F, SANDHU R. An attribute-based access control extension for openstack and its enforcement utilizing the policy machine[C]//IEEE, International Conference on Collaboration and Internet Computing. 2017:37-45.

- [14] 王于丁, 杨家海. 一种基于角色和属性的云计算数据访问控制模型[J]. 清华大学学报(自然科学版), 2017, 57(11): 1150-1158.

WANG Y D, YANG J H. Data access control model based on data's role and attributes for cloud computing[J]. Journal of Tsinghua University (Natural Science Edition), 2017, 57(11): 1150-1158.

- [15] GUPTA M, PATWA F, SANDHU R. Object-tagged RBAC model for the Hadoop ecosystem[C]//IFIP Conference on Data and Applications Security and Privacy. 2017: 63-81.

[作者简介]



苏秋月(1993-), 女, 四川绵阳人, 四川大学硕士生, 主要研究方向为大数据平台数据治理与访问控制。



陈兴蜀(1968-), 女, 贵州六枝人, 博士, 四川大学教授、博士生导师, 主要研究方向为云计算与大数据安全、可信计算与信息保障。



罗永刚(1980-), 男, 贵州瓮安人, 博士, 四川大学助理研究员, 主要研究方向为云计算及大数据安全。