文章编号:1000－2367(2016)05－0146－08 DOI:10.16366/j.cnki.1000－2367.2016.05.025

一种适用于 Hadoop 平台的基于属性访问控制模型

陈垚坤,刘文丽

(江南计算技术研究所,江苏 无锡 214083)

摘 要:针对 Hadoop 平台缺乏有效访问控制机制的问题,提出一种适用于 Hadoop 平台的基于属性访问控制模型 H-ABAC.该模型将传统 ABAC 模型扩充为五元组,加入安全等级属性增加了灵活性,选择 XACML 为策略描述语言并提供标准化、可大规模扩展的访问控制策略.对该模型进行形式化定义,构建模型框架并详述各个模块的功能与实现,对模型的适用性和优势进行了分析.分析得出:该模型可以满足自主、细粒度以及动态授权的需求.仿真实验显示:H-ABAC 可以有效控制策略数量并且减少系统的开销,所增加时间开销也在可控范围之内.

关键词:Hadoop;访问控制;基于属性;XACML;细粒度

中图分类号:TP393 **文献标志码**:A

随着大数据时代的到来,商务、政务及制造业等一大批行业向数据化与信息化极速迈进.与此同时,大数据的安全问题也逐渐显露出来.根据 Gartner 的预测,传统的数据安全解决方案到 2020 年将会失效,到 2016 年,25%的大型企业将会采用至少一种大数据安全解决方案.而随着数据安全的威胁愈加严重复杂,与之而来的严重后果也日益引起各方关注.可以说,大数据安全已经上升成为国家安全极为关键的组成部分,已成为国家最重要的战略安全之一.

Hadoop 最早起源于 Apache 开源搜索引擎 Nutch,2006 年从 Nutch 分离出来,其后加入了众多子项目[1－2].Hadoop 以分布式文件系统 HDFS、并行计算模型 MapReduce 以及分布式存储系统 HBase 为核心,提供一个系统底层细节透明的分布式基础框架[3－4].Hadoop 平台凭借其高可靠性、高扩展性、高效性及高容错性成为目前最流行的大数据处理平台之一.Hadoop 已经在大批互联网企业得到成功应用,在国内外产业界、学术界都受到广泛关注.华为在 Hadoop 框架的基础上实现了其"弹性大数据"的应用服务,用于海量数据的管理和分析;Yahoo 利用 Hadoop 集群支持广告系统和 Web 搜索的研究;Facebook 借助 Hadoop 集群来支持其数据分析和机器学习;百度使用 Hadoop 进行网页数据挖掘和搜索日志分析工作;阿里利用 Hadoop 集群存储并处理电子商务交易的相关数据.然而,作为常用的大数据处理平台,Hadoop 提供的安全机制并不完善,尤其是大数据场景下用户和资源的规模庞大及动态性使得 Hadoop 现有的访问权限管理非常复杂,不能满足应用安全需求,严重地制约了大数据及 Hadoop 的发展与推广.因此,为 Hadoop 平台提出一种能够满足大数据环境下自主、动态、细粒度的访问控制模型成为重要研究课题.

1 相关研究

早期的 Hadoop 没有任何系统安全机制,第一代 Hadoop 从 0.20.0 版本开始加入基于 Kerberos 的身份认证机制[5].用户通过身份认证之后,可以向 Hadoop 集群的主节点申请数据服务或者提交作业.权限授予之后不再监管,无法阻止合法用户的非法操作.第二代 Hadoop 的安全机制包括身份认证和访问控制:身份认证采用 Kerberos 和 Token 两种方案,Client-NameNode 之间初次通信采用 Kerberos 进行身份认证,之后便换用较小开销的 Delegation Token[6],而 DataNode-NameNode 之间的认证始终采用 Kerberos 机制[7];访

收稿日期:2016-04-17;**修回日期**:2016-06-21.

基金项目:国家核高基项目 (2013ZX01029002-001)

第 1 作者简介(通信作者):陈垚坤(1991－),男,陕西洋县人,江南计算技术研究所硕士研究生,研究方向为信息安全,E-mail:pkucyk@163.com.

问控制授权是通过访问控制列表 ACL(Access Control List)实现的，按照授权实体，可分为队列访问控制列表、应用程序访问控制列表和服务访问控制列表. 在一次访问过程中，首先通过 Kerberos 服务器对 Client 进行身份认证，确定 Client 身份，然后检查访问控制列表，看 Client 是否有访问资源或提交作业的权限. 一旦访问者通过验证，会获取 HDFS 或者 MapReduce 授予的 delegation token，之后的任何操作，均要检查该 token 是否存在，且使用者是否和之前注册使用该 token 的人一致.

Hadoop 平台目前采用的访问控制列表属于自主访问控制 DAC，自主访问控制本身安全性能并不高，且不能满足大数据环境下的安全需求. 在大数据环境下，因为数据规模大且动态性较强，使访问控制列表变得规模巨大且不易维护，权限管理复杂困难，因此迫切需要找到一种能适应大数据环境特点的访问控制机制. 文献[8]把 BLP 模型与 Biba 模型有机结合，提出一种基于行为的云计算访问控制模型 CCACSM，该模型能保证数据的完整性和保密性，但其继承于 BLP 模型的“上读下写”特性使可用性降低. 文献[9]将主客体安全等级和动态可变机制引入进访问控制策略，提出一种基于角色的访问控制模型 CCRBAC，该模型满足了云计算环境下访问控制安全性、灵活性和可靠性的需求，但对于用户的权限并不进行监管和控制. 文献[10]提出一种基于信任的访问控制模型 LT，为用户设定信任值并根据行为修改信任值，通过信任值对用户动态控制，该模型对用户行为进行监管，解决了权限授予后不再监管的问题，但没有考虑上下文环境对权限授予的影响.

本文根据以上文献的研究成果，以基于属性的访问控制模型为基础，提出一种适用于 Hadoop 的访问控制模型. 该模型借鉴主客体安全等级和动态控制的思想，充分考虑上下文环境对授权的影响，制定包含主体、客体、操作、环境和安全等级五元组的属性策略，引入 XACML 作为模型的策略描述语言，并加入对属性变化的监控模块，可以满足自主、细粒度和动态授权的安全需求.

2　背景知识

2.1　经典的基于属性的访问控制模型 ABAC

基于属性的访问控制模型 ABAC(Attribute Based Access Control)是一个四元组(S,O,P,E)，其中 S，O，P 和 E 分别是由主体属性、客体属性、操作属性和环境属性确定的主体、客体、操作和环境集合[11]. ABAC 制定完善的属性策略集，用户的每一次访问都要通过策略集判定是否合法，若合法则授予相应的访问权限.

传统访问控制模型包括自主访问控制 DAC、强制访问控制 MAC、基于角色访问控制 RBAC. 这些模型在授权时忽略了上下文环境的影响，属于静态授权机制. 此外 DAC 难以应对大数据环境下用户资源的巨大规模及动态性，访问控制列表 ACL 会相应急剧扩大；MAC 难以定义安全密级且无法进行细粒度的访问控制；RBAC 需要定义大量的角色及所对应权限，角色、权限的管理和维护变得很困难. 与传统访问控制模型相比，ABAC 可以增删修改属性类型，将一些被忽略的元素例如上下文环境等加入模型；可以在属性策略中随意组合各个属性，利于细粒度的访问控制；还可以灵活地修改属性策略中每个属性的可取值范围，以应对用户资源的动态性. 综上 ABAC 能够动态扩展且较容易实现细粒度访问控制，适合作为基础模型将其改进后应用于 Hadoop 平台.

2.2　XACML 概述

可扩展访问控制标记语言 XACML(Extensible Access Control Markup Language)是一种通用的策略语言和访问决策语言[12]. XACML 与 ABAC 存在很多的共同点，是实现 ABAC 模型的主要手段. 基于 XACML 的 ABAC 模型包括 PAP(策略管理点)、PIP(策略信息点)、PDP(策略决策点)和 PEP(策略执行点)等 4 个组成部分，在分布式的环境中可以根据 ABAC 各个元素的属性动态地评估访问请求，并进行授权决策，如图 1 所示.

模型各组成部分的主要功能如下：PAP 存放 XACML 语言编写的属性策略集，供 PDP 使用；PEP 接受来自用户的访问请求，发送授权请求给 PDP，接受 PDP 的决策并执行；PDP 利用属性策略集来判断用户的访问请求是否合法，并将决策结果返回给 PEP；PIP 负责给 PDP 提供授权决策所需要的属性信息.

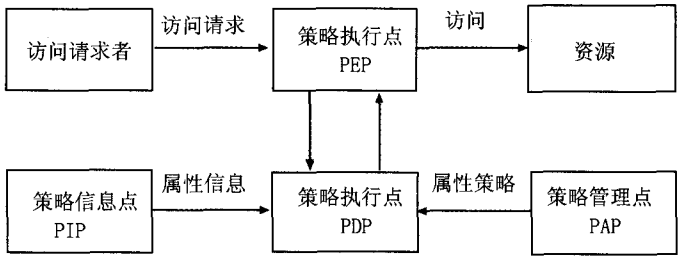

图1 基于XACML的ABAC模型

3 基于属性的 Hadoop 访问控制模型 H-ABAC

Hadoop 未来会在 IT、军事、政府或其他公共部门有广阔的应用场景. 军方政府等保密部门对于数据的安全保密性有着严格的要求,在一些场景中,涉密程度较低的人员无论任何条件下都无法接触应用系统中的某些核心机密;而随着商业秘密越来越被重视,商业部门也存在相同的需求. 在这种情况下,为访问者增加密级标签,将密级标签融合入访问控制模型,规范为密级属性,再制定一条关于密级的策略,直接限制低密级的主体访问该资源,即可满足访问控制要求;如果采用经典 ABAC 四元组制定策略,则可能需要制定数条甚至数十条策略,限制平台中所有无权访问者,这样将使访问控制变得复杂且代价大.

在 Hadoop 中,可以根据数据所对应的敏感程度,采取不同的存储方式. 引入密级属性后,密级为公开的数据进行明文存储,其余密级的敏感数据进行加密存储. 加密存储的敏感数据中,其敏感程度也有所不同,可以进行基于属性的分级加密. 在 Hadoop 平台对数据进行分级加密的情况下,可以制定关于其他属性与密级属性相结合的策略,控制加密数据密钥的发放. 即:某一数据按照机密级进行加密,则可以制定策略,规定本次访问主体属性、环境属性等满足一定要求且密级属性必须为机密级以上才可获得密钥.

密级属性和其他属性之间存在关联. 以环境属性为例,表现在相同的主体在不同环境属性值下拥有的密级属性值是不同的. 例如某政府单位的用户,在访问本单位所属数据和访问某外单位所属数据时,其环境属性中 IP 属性分别为内网 IP 和外网 IP,其所具有的密级级别,前者要高于等于后者. 用户的身份在一定程度上可视为一系列的属性组合,密级属性可视为用户身份的一个重要特征. 用户首次访问,可由权威的身份认证系统在初始身份认证时结合用户的主体属性为其制定初始的密级,之后密级属性根据其他属性的变化而进行相应的修改.

基于以上原因,可以看出,密级元素可以很好地和主体、客体、环境等属性结合制定策略,也可自身单独形成一条策略,可以使访问控制策略更细化且更灵活多样化.

3.1 模型形式化定义

本文以经典的 ABAC 模型为基础,在访问控制策略中引入密级元素,将密级元素作为模型中策略评估的一个标准. 将访问控制策略从主体、客体、操作和环境四元组扩充为主体、客体、操作、环境和密级五元组.

定义 1 基于属性的 Hadoop 访问控制模型 H-ABAC:H-ABAC 定义为五元组(S,O,P,E,L),S 为主体,O 为客体,P 为操作,E 为环境,L 为密级.

定义 2 访问请求 Request:Request=Req. s⊗Req. o⊗Req. p⊗Req. e⊗Req. l,其中

Req. s={<Sattr1, value>, <Sattr2, value>,…, <Sattrm, value>},表示主体具有的属性名值对集合;

Req. o={<Oattr1, value>, <Oattr2, value>,…, <Oattrn, value>},表示客体具有的属性名值对集合;

Req. p={<Pattr, value>},表示操作的属性名值对,其中 value 的值域为{R, W, RW, EXE};

Req. e={<Eattr1, value>, <Eattr2, value>,…, <Eattrk, value>},表示环境具有的属性名值对集合;

Req. l={<Lattr, value>},表示密级的属性名值对,其中 value 的值域为{public, secret, confidential, topsec},分别对应公开、秘密、机密、绝密,可根据实际需要进行扩展.

一个典型的 H-ABAC 模型策略结构如图 2 所示为树形结构，其中访问控制策略 Policy 和策略集 Policyset 都可作为根节点，而访问控制规则 Rule 作为叶子节点.

定义 3　访问控制策略 Policy/策略集 Policyset：Policy 和 Policyset 是 H-ABAC 中存储访问控制策略的基本单位，Policyset 可包含多条 Policy，Policy 可包含多条访问控制规则 Rule.

定义 4　访问控制规则 Rule：Rule 由 effect 和 target 两个元素组成，其中 effect 表示本条规则的授权判定结果，取值范围为{permit，deny}，分别对应允许访问、拒绝访问.

定义 5　目标 target：target 存在于策略树的三级结构中，用来判断一条访问请求 Request 是否能与本规则/策略/策略集匹配.

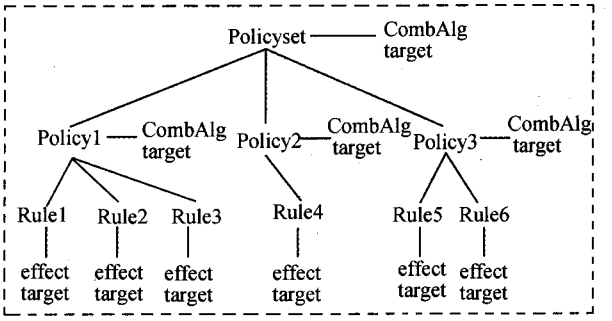

图2 H-ABAC模型策略树结构

规则 Rule 中的 target 是与 Request 进行匹配的最小策略单元，若 Rule 中的 target 能与 Request 匹配（即 Request 的属性集合能包含于 target 的属性集合），则 Rule 对 Request 判定结果为 effect，否则为 notmatch. Rule 中的 targe 定义为 target＝tar. s⊗tar. o⊗tar. p⊗tar. e⊗tar. l，用 DOM 表示主体各属性的值域，则 tar. s＝{＜Sattr1，DOM＞，＜Sattr2，DOM＞，…，＜Sattr$m$，DOM＞}，表示主体各属性可取值范围的集合，tar. o，tar. p，tar. e，tar. l 的表述可由 tar. s 类推.

策略 Policy 中的 target 是其所属所有 Rule 中 target 的并集，与之类似策略集 Policyset 中的 target 是其所属所有 Policy 中 target 的并集. 两个级别的 target 与 Request 匹配，若匹配成功，则说明本策略/策略集可以适用 Request，否则不适用.

定义 6　策略合并算法 CombAlg：Policy 中如果有多条结果不同的 Rule，一些 Rule 允许访问，另外的 Rule 拒绝，就需要 CombAlg 解决冲突，对不同 Rule 的结果进行合并. Policyset 的 CombAlg 与之同理. XACML 中有 4 种组合算法，即拒绝优先算法、许可优先算法、首先应用算法和唯一应用算法.

3.2　模型框架与主要功能模块

H-ABAC 模型框架如图 3 所示.

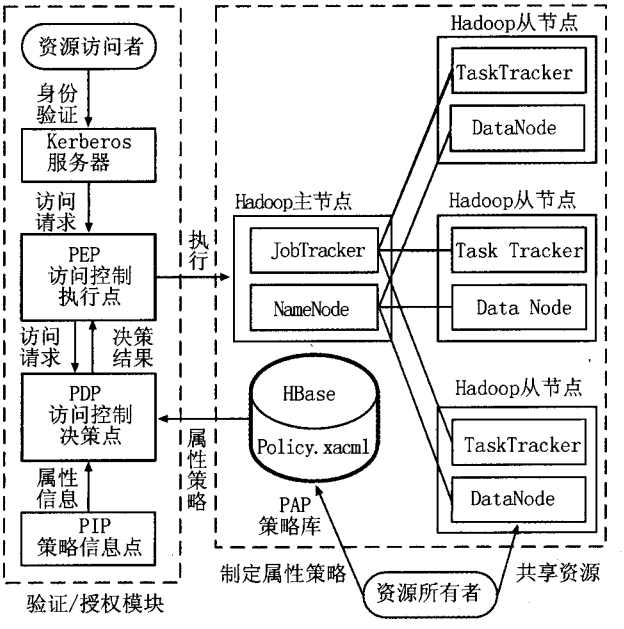

图3 H-ABAC模型框架

(1) PAP 模块功能与实现

PAP 是 H-ABAC 模型中的策略库,主要任务是为 PDP 匹配适用的属性策略,由定义 3、定义 4 可知,模型的策略以 xacml 格式存储在 PAP 中,为树形结构,策略匹配的过程即为策略树剪枝的过程. 访问请求 Request 首先与根节点 Policyset 中 target 匹配,由定义 5 可知,根节点 Policyset 中 target 是其所有叶子节点 target 的并集,若不能匹配,则本策略树全部剪枝;若可以匹配,进一步使用广度优先搜索的方式与子节点及至叶子节点进行匹配,并对不可匹配的 Policy 与 Rule 进行剪枝. 本文为 PAP 提供策略匹配算法如下.

算法 1 策略匹配算法 PolicyMatch

```
输入:REQ——访问请求
     PolicySet——策略集
输出:Rule_Effects——策略集中所有可匹配规则判定结果集合
all Rule_Effects←not-match
if REQ∈PolicySet.target//Policyset 一级的剪枝
  for each policy[i](i∈n) in PolicySet
    if REQ∈Policy[i].target//Policy 一级的剪枝
      for each Rule[j](j∈n) in policy[i]
        if REQ∈Rule[j].target//Rule 一级的剪枝
          Rule_Effects[Policy[i]-Rule[j]]=Rule[j].effect
          end if
        end for
      end if
      else continue next policy
  end for
else return PolicySet-not-match
```

在一些较大的应用项目中,系统内可能存在几十万到几百万条策略,可将策略存储在 Hadoop 平台的分布式数据库 HBase 中,并对策略建立高效的缓存机制或者索引,可大大提高策略匹配的速率与效率.

(2) PDP 模块功能与实现

PDP 是 H-ABAC 模型中的策略决策模块,主要任务是用 PAP 中匹配的策略集 Policyset 计算合并得到访问请求 Request 的判定结果并将其返回给 PEP. 本文为 PDP 提供策略匹配算法,如下:

算法 2 结果判定算法 ResultEvalucation

```
输入:Rule_Effects——算法 1 所得
输出:REQ_RES——对本次访问请求的判定结果
for each policy[i](i∈n) in PolicySet
  for each Rule[j](j∈n) in policy[i]
      //所有可匹配的 Rule 在其 Policy 内部合并得出结果
      if Rule_Effects[Policy[i]_Rule[j]] ! =not-match
        Combine(AllChild(Policy[i]), PolicyCombAlg[i])
      break
    end if
  end for
end for
//将所有 Policy 结果合并得到 Policyset 的结果
REQ_RES←Combine(AllChild(Policyset),
                    PolicysetCombAlg)
return REQ_RES
```

(3) PEP 模块功能与实现

PEP 是 H-ABAC 模型中的策略执行模块，主要任务是接受访问请求 Request，将其发送给 PDP，接受 PDP 的决策结果并执行.

(4) PIP 模块功能与实现

PIP 负责给 PDP 提供决策所需要的属性信息，PIP 中维护主体、客体、操作、环境、密级属性信息表. 与 PAP 类似，在属性信息表规模较大的情况下，可建立高效的缓存机制来提高查找速率与效率.

3.3 模型访问请求判定过程

在 H-ABAC 模型中，资源访问者 Client 一次访问请求判定过程包括以下步骤.

步骤 1 Client 通过 Kerberos 进行身份认证.

步骤 2 Client 通过身份认证后，向 PEP 发送访问请求 Request.

步骤 3 PEP 将 Request 发送至 PDP 准备进行决策.

步骤 4 PDP 搜集 Request 决策需要的所有属性信息，分为两部分：1)PDP 使用 xacml 提供的两种机制 AttributeDesignator 和 AttributeSelector 来解析来自 Request 中的属性信息；2)PDP 向 PIP 发命令，由 PIP 在其存储的属性表中收集相关的属性信息.

步骤 5 PDP 得到处理 Request 需要的所有属性信息，扩展组合成为 Request (req. s, req. o, req. p, req. e, req. l)，并将其转化为 xacml 格式便于处理.

步骤 6 PDP 向 PAP 发送策略匹配请求，以找到适合判定的策略集.

步骤 7 PAP 使用算法 1 在策略库中查找匹配的策略集，将所有适合判定的策略集 Policysets 发送至 PDP.

步骤 8 PDP 使用算法 2 对 Policysets 进行计算与合并.

步骤 9 PDP 通过计算合并得到最终判定结果 REQ_RES，若 REQ_RES 为 permit，则表示用户 req. s 在条件状态为 req. e 和密级为 req. l 的情况下，请求对资源 req. o 进行 req. p 操作被允许，否则表示该请求被拒绝.

步骤 10 PDP 将判定结果返回给 PEP.

步骤 11 PEP 将判定结果返回给资源访问者. 资源访问者根据判定结果访问 HDFS 中的数据资源或者向 MapReduce 提交作业.

3.4 模型动态监控机制

在访问过程中，有可能出现资源、环境动态变化的情况，而此时原来的决策结果不一定适用，需要终止本次访问并重新进行访问请求. 本文在 H-ABAC 中设置监控模块处理以下情况：

1) 上下文环境出现变化，监测到环境发生变化，终止本次访问，并向 PIP 发信息更新环境属性和密级属性，更新完成后由访问者重新发起访问请求；

2) 资源所有者修改资源，监测到资源发生变化，终止本次访问，提醒资源所有者根据资源变化来修改所对应的属性策略，并向 PIP 发信息更新客体属性，待 PAP 属性策略与 PIP 资源属性信息更新完成后由访问者重新发起访问请求.

3.5 模型优势分析

H-ABAC 模型吸收了传统访问控制模型的优点，充分考虑了分布式动态环境对授权的影响，其优势主要体现在以下几个方面.

1) 细粒度访问控制. H-ABAC 模型将一次访问过程的所有元素使用属性描述，可以严格控制访问者取得权限的各种条件并且将可访问范围精确到 Hadoop 中 DataNode 块级别的资源，满足最小权限原则. 将密级属性这一重要因素引入到访问控制策略中，对访问控制策略进行进一步的细化. 模型中策略描述语言 xacml 提供了多种策略合并算法，使得策略的合成更加灵活多变，访问控制策略表达能力更强，策略表达能力越强，说明其控制粒度越细.

2) 自主授权. 在 H-ABAC 模型中，所有的访问控制策略都由资源所有者制定，且策略上传之后，资源所有者可以根据自身实际应用需求修改策略，保证了资源能够按照其所有者的意愿被访问.

3) 动态访问控制. 在 H-ABAC 模型中,属性的定义和设置有很大的灵活性,能够大规模动态扩展,可以满足各种规模的应用系统的需求. 模型中设置了监控模块,针对各类属性动态变化的情况,都制定了应对机制,可以及时中断访问,避免因属性动态变化造成的非法访问,保证了模型的安全性.

4) 较低的系统复杂性与较小的系统开销. 传统访问控制在用户和资源数量大幅度增加的情形下,访问控制规则数目会出现指数级的增长,系统的维护量急剧增加,加大了系统的开销. H-ABAC 采用的(S,O,P,E,L)五元组比经典的 ABAC(S,O,P,E)四元组能更有效的控制策略的规模,因为在相同情况下,一条关于密级元素的规则被四元组规则替换往往需要增加几条甚至几十条规则. H-ABAC 中,访问控制规则随用户和资源数量的增长只是表现出线性的增加,从而降低了系统的复杂性与系统开销.

4 实验仿真

在实验室搭建 Hadoop 平台,其物理配置为一个主节点和 6 个从节点. 主节点部署在服务器上,配置为 Intel(R) Xeon(TM) E5-2600v3 CPU,64 G 内存,100 Mbps 以太网,8 T 硬盘;从节点部署在 PC 上,配置为 Intel(R) Core(TM) i5-2500 CPU,4 G 内存,100 Mbps 以太网,500 G 硬盘.

在平台中加入 H-ABAC 模型,部署一定数量的资源,并为其制定相应的策略. 因为考虑到现实中存储到云平台中的结构化数据、半/非结构化数据所占的比率分别是 5%～15%、85%～95%,因此在上传资源时总比例也保持在这个比率,其中半结构化数据为 XML,HTML 等格式的数据,非结构化数据包括 word 文档、视频、音频、图片等等. 为验证 H-ABAC 能有效控制策略总数,将 H-ABAC 中与密级元素相关的规则替换为 ABAC 四元组规则,并比较其数量与增长趋势,图 4 统计了资源数量增加的情况下,四元组 ABAC 和五元组 H-ABAC 策略库中规则总数的变化曲线:

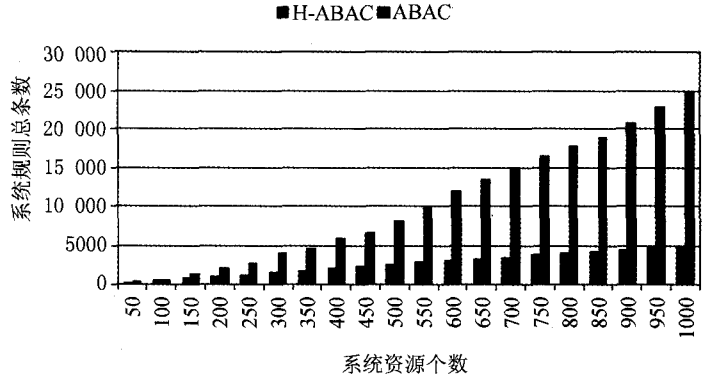

图4 资源数量与规则总数的关系

由图 4 可知 H-ABAC 的规则增长趋势明显缓于 ABAC 的增长趋势,因此 H-ABAC 能有效控制策略总数,更适合应用于用户和资源数量都很庞大的系统.

加入了 H-ABAC 的 Hadoop 平台与普通 Hadoop 平台访问耗时主要差异在 PDP 模块,在平台 PDP 模块中加入计时器,统计不同策略数量下 PDP 模块耗时,结果如图 5 所示.

由图 5 可知随着访问控制策略的增加,PDP 决策时间缓慢上升,在 1500 条策略以内,增加的时间开销在 1 s 以内,时间开销较小. 但在大型系统中,策略数量可达到几万条甚至几十万上百万条,此时 PDP 的决策时间将成为一个较大的开销,

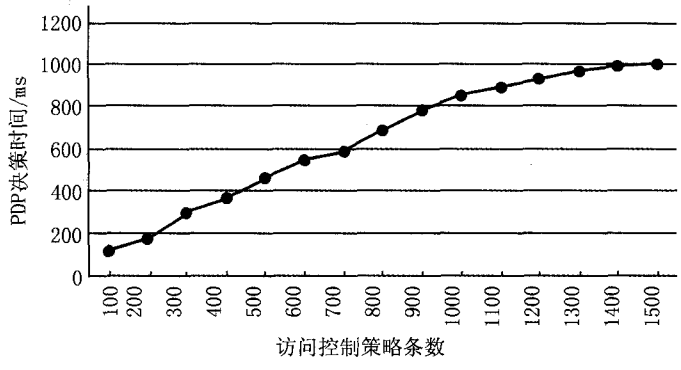

图5 访问控制策略数量与PDP决策时间的关系

可以采取建立高效缓存和索引的方式，有效减少开销，使其保持在一个可控范围之内.

5 结　论

本文基于经典的 ABAC 模型提出一种适用于 Hadoop 平台的基于属性访问控制模型. 将密级元素扩展入模型，增加了属性策略表达的灵活性与完备性，同时结合 XACML 实现了模型中各个模块. 分析及实验证明本模型能够实现自主、细粒度以及动态授权. 在未来的工作中，将继续研究属性策略库的高效缓存和索引机制，使模型在大规模的应用系统中也能保持较小的开销.

参 考 文 献

[1] Nandimath J, Banerjee E, Patil A, et al. Big data analysis using Apache Hadoop[C]. Information Reuse and Integration (IRI), 2013 IEEE 14th International Conference on, Milan, 2013.

[2] 张欣晨，杨　庚. Hadoop 环境中基于属性和定长密文的访问控制方法[J]. 计算机工程与应用，2015，51(23)：87-93.

[3] Liu Y, Chen B, He W, et al. Massive image data management using HBase and MapReduce[C]. International Conference on Geoinformatics, Kaohsiung, 2013.

[4] Taylor R C. An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics[J]. Bmc Bioinformatics, 2010, 11(6): 3395-3407.

[5] Zheng K, Jiang W. A token authentication solution for hadoop based on kerberos pre-authentication[C]. Data Science and Advanced Analytics (DSAA), Shanghai, 2014.

[6] Kim S H, Lee I Y. Block Access Token Renewal Scheme Based on Secret Sharing in Apache Hadoop[J]. Entropy, 2014, 16(8): 4185-4198.

[7] Murthy A C, Vavilapalli V K, Eadline D, et al. Apache Hadoop YARN: Moving Beyond MapReduce and Batch Processing with Apache Hadoop 2[J]. Pearson Schweiz Ag, 2014, 2(5): 56-59.

[8] 林果园，贺　珊，黄　皓，等. 基于行为的云计算访问控制模型安全模型[J]. 通信学报，2012，33(3)：59-66.

[9] 赵明斌，姚志强. 基于 RBAC 的云计算访问控制模型[J]. 计算机应用，2012，32(2)：267-270.

[10] 刘　莎，谭　良. Hadoop 云平台中基于信任的访问控制模型[J]. 计算机科学，2014，41(5)：155-163.

[11] 王小明，付　红，张立臣. 基于属性的访问控制研究进展[J]. 电子学报，2010，38(7)：1660-1667.

[12] Turkmen F, Hartog J D, Ranise S, et al. Analysis of XACML Policies with SMT[M]. Berlin: Springer, 2015.

Attribute-based Access Control Model for Hadoop

CHEN Yaokun, LIU Wenli

(Jiangnan Computing Technology Research Institute, Wuxi 214083, China)

Abstract: An attribute-based access control model for Hadoop(H-ABAC) is proposed to solve the access control problem in Hadoop. The traditional ABAC model is expanded to five elements. The security level is considered as an important element for H-ABAC like subject, object, operation and environment. Standardized and extensible access control policies are evolved by XACML. Modules of H-ABAC are formally defined. The functions and implementation of these modules are detailedly described. The applicabilities and superiorities of H-ABAC are analysed. The conclusion shows that H-ABAC can provide independent, fine-grained and dynamic access control and Reduce the system overhead. The simulation experiment shows that H-ABAC can keep the amount of access control policies slowly increasing and the cost of time is acceptable. All that shows H-ABAC is a practical access control model for Hadoop.

Keywords: Hadoop; access control; attribute-based; XACML; fine-grained