面向 HDFS 的密钥资源控制机制

金伟[1,2,3]，李凤华[1,2]，余铭洁[1,4]，郭云川[1,2]，周紫妍[1,2]，房梁[1]

（1. 中国科学院信息工程研究所，北京 100093；2. 中国科学院大学网络空间安全学院，北京 100049；
3. 中国信息通信研究院，北京 100191；4. 中国科学技术大学网络空间安全学院，安徽 合肥 230027）

摘 要：大数据环境呈现多用户跨网交叉访问、多服务协同计算、数据跨服务流动、海量文件管控复杂的特点，现有密钥资源控制模型和机制不完全适用于大数据场景。针对大数据环境的密钥资源控制、操作语义归一化描述、细粒度访问控制的需求，从密钥资源控制的场景要素及属性出发，通过映射面向网络空间的访问控制（CoAC）模型，提出了面向 HDFS 的密钥资源控制机制；然后，给出了面向 HDFS 的密钥资源控制管理机制（CKCM），包括管理子模型和管理支撑模型，并用 Z 语言形式化地描述了管理模型中的管理函数和管理方法；最后，基于 XACML 实现 CKCM 系统，实现 HDFS 中密钥及文件资源的细粒度安全访问控制。

关键词：大数据平台；密钥管理；资源控制；面向网络空间的访问控制

中图分类号：TP302

文献标志码：A

DOI: 10.11959/j.issn.1000−436x.2022165

HDFS-oriented cryptographic key resource control mechanism

JIN Wei[1,2,3], LI Fenghua[1,2], YU Mingjie[1,4], GUO Yunchuan[1,2], ZHOU Ziyan[1,2], FANG Liang[1]

1. Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
2. School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China
3. China Academy of Information and Communications Technology, Beijing 100191, China
4. School of Cyber Security, University of Science and Technology of China, Hefei 230027, China

Abstract: The big data environment presents the characteristics of multi-user cross-network cross-access, multi-service collaborative computing, cross-service data flow, and complex management of massive files. The existing access control models and mechanisms are not fully applicable for big data scenarios. In response to the needs of fine-grained access control and multi-service strategy normalization for cryptographic data in the big data environment, starting from the scene elements and attributes of access control, the HDFS-oriented CKCM was proposed by mapping the cyberspace-oriented access control (CoAC) model. Subsequently, a fine-grained access control management model for HDFS was proposed, including management sub-models and management supporting models. The Z-notation was used to formally describe the management functions and management methods in the management model. Finally, the CKCM system was implemented based on XACML to realize fine-grained secure access control for managing file and secret keys in HDFS.

Keywords: big data platform, cryptographic key management, resource control, cyberspace-oriented access control

收稿日期：2022−03−07；**修回日期**：2022−06−07

通信作者：郭云川，guoyunchuan@iie.ac.cn

基金项目：国家自然科学基金资助项目（No.U1836203, No.61872441）；国家重点研发计划基金资助项目（No.2018YFB2100400）；中国科学院青年创新促进会人才基金资助项目（No.2021154）

Foundation Items: The National Natural Science Foundation of China (No.U1836203, No.61872441), The National Key Research and Development Program of China (No.2018YFB2100400), The Youth Innovation Promotion Association of Chinese Academy of Sciences (No.2021154)

0 引言

Hadoop 分布式文件系统（HDFS, Hadoop distributed file system）是大数据平台 Hadoop 集群下对海量数据进行分布式存储的大数据文件存储系统，为 Hadoop 集群上的数据和组件提供高可用数据存储服务。目前，HDFS 存储系统在物联网数据中心等各类数据分析和存储场景下具有广泛应用。

据《物联网终端安全白皮书（2019）》统计，截至 2019 年，全球物联网设备连接数量达 110 亿个，我国授权频段蜂窝物联网终端连接数量达 9.2 亿个。中国互联网络信息中心发布的《第 48 次中国互联网络发展状况统计报告》显示，2021 年上半年，移动互联网接入流量达 1 033 亿 GB。庞大的终端账户规模和数据量需要大数据平台进行数据管理和安全防护。同时，敏感设备如终端支付设备等，涉及用户身份、位置隐私、信用卡账户、生物识别特征等核心隐私信息，一旦泄露，将威胁千万用户的财产安全，需要 HDFS 采用访问控制、数据加密等手段进行保护，提供安全的存储服务。

现有 HDFS 系统具备基本的资源控制能力，但其控制的客体对象仅针对 HDFS 的目录和文件，且大部分 HDFS 文件为明文存储，访问控制的安全性依赖策略中对权限的配置。一旦权限配置不当或访问控制被绕过，将造成明文数据泄露，依然会威胁数据安全。

通过设置加密区保护目录和文件，加密区密钥在密钥管理系统（KMS, key management system）中单独存储和访问，并非实时计算，且未与文件统一访问控制，存在密钥库泄露的风险。因此需对密钥进行访问控制，否则，倘若敌手经其他途径可以拿到分布式存储的数据，那么只要获取密钥，就可以解密数据、拿到明文，损害文件的机密性；同时，若敌手可以单独销毁密钥，导致密文无法解密，也将破坏文件的可用性。现有 KMS 的访问控制机制非常简单，只提供黑白名单匹配的方式进行控制，管理效率低。因此，需同时对文件数据和密钥实施访问控制。

现有 HDFS 的密钥资源控制及访问控制模型粒度较粗，HDFS 原生的 POSIX 权限模型及 POSIX ACL 机制只从用户组的读、写、执行等基本操作进行访问控制；Apache Sentry 为 HDFS 提供基于角色的访问控制；Apache Ranger 为 HDFS 提供基于标签（角色的泛化）的访问控制。已有组件都不提供基于时间、IP 地址等实时环境，以及细粒度主客体属性的访问控制。因此需要设计完整高效的细粒度访问控制机制，提供数据及密钥的全生命周期保护。

根据上述现状和安全特点，大数据安全防护需要满足以下新需求。1) 密钥资源控制：集中存储和管理的密钥需要有效的访问控制，否则仍可能导致数据泄露或不可用。2) 操作语义归一化描述：HDFS 所支撑的服务众多，包括 HDFS 文件系统本身、KMS 密钥管理、加密区管理等服务，访问控制策略需要归一化描述和鉴权，且需要统一的管理函数。3)细粒度访问控制：细粒度大数据访问控制系统应具备对于文件和密钥的细粒度访问控制，对于谁、在什么时间、在什么场景下、从何种设备、经由何网络、通过何操作、访问何数据等问题，应设置属性和策略进行保护。

文献[1]为了实现网络数据的细粒度控制，通过考虑访问请求实体、广义时态、接入点、访问设备、网络、资源、网络交互图和资源传播链等要素，提出了面向网络空间的访问控制（CoAC, cyberspace-oriented access control）模型。本文将 CoAC 映射到 HDFS 的文件与密钥管理中，实现面向 HDFS 的密钥资源控制机制（CKCM, cryptographic key control mechanism）。该机制将 CoAC 中的访问请求实体、资源、广义时态、接入点、访问设备、网络等要素具体化为 HDFS 中的多服务实体，作为大数据环境下的访问控制模型实例，管理大数据文件与密钥资源。CoAC 模型的宏观架构清晰、控制要素丰富，与大数据环境下的安全需求相匹配，具有一致性。本文的主要贡献如下。

1) 针对大数据环境对密钥资源控制、细粒度访问控制等需求，通过映射面向网络空间的访问控制模型，提出了面向 HDFS 的密钥资源控制机制，实现文件与密钥统一访问控制。

2) 针对大数据策略管理的复杂性和操作语义归一化描述的安全需求，给出了面向 HDFS 的密钥资源控制管理模型，并用 Z 语言形式化地描述了管理模型中的管理函数和管理方法。

1 相关工作

大数据环境下，密钥的访问控制机制为黑白名单，其余相关工作较少。因此本文扩展至大数

据访问控制角度进行梳理，并介绍其他细粒度访问控制模型。大数据平台中，各类访问控制模型、机制、工具随着技术发展而不断演进，主要相关工作如下。

1.1　大数据平台的访问控制模型

大数据场景数据量大且多源多样，Colombo 等[2]从 MapReduce 的分布式计算场景，梳理了可用的访问控制模型，包括 GuardMR[3]、Vigiles[4]、HeAC[5]、OT-RBAC[6]等。其中，以 HeAC 和 OT-RBAC 等最为贴合大数据存储的使用场景，讨论如下。

Gupta 等[5]首先介绍了目前 Hadoop 生态系统中的用户级、服务级、数据级、资源级等多层授权机制和相关组件，随后提出了 HeAC（access control for Hadoop ecosystem）模型。HeAC 是一个形式化的 Hadoop 多级访问控制模型，是关于 Apache Ranger、Sentry 和原生 Hadoop（native Apache Hadoop）等组件中访问控制能力的模型抽象。该模型描述了 Hadoop 服务、生态服务系统（如 Hive、Kafka 等组件）中数据的授权模型和读取访问控制机制。基于标签对 Hadoop 中的各类操作进行访问控制，对于生态服务客体如 Hive 等，可以设置属性值作为标签（Tag）用于赋权，并结合 NIST 的 RBAC96[7]对模型进行修改。但该模型未考虑上传操作，且仅考虑了允许的权限，未考虑拒绝的判定处理。

随后，Gupta 等[6]扩展 HeAC 至 OT-RBAC（object-tagged role-based access control）模型，引入组继承（GH, group hierarchy）机制。并且扩展 Hadoop 服务的权限也可以分配给角色 R；结合组继承机制，可以获得每个用户/主体/角色的有效角色。该模型在 Apache Ranger 0.5 的基础上实现。

Gupta 等[8]引入信任的概念和基于属性的访问控制，将 HeAC 模型扩展为 HeABAC 模型，分析了多服务间数据信任的反射性、传递性、对称性，给出了形式化定义、授权函数、决策函数和管理模型，并给出了用于 IoT 场景中的实例。

Awaysheh 等[9]针对 Apache Hadoop 3.x 梳理了联邦 HDFS 中的访问控制场景，在联邦 HDFS 多 NN 联合管理、去中心化多服务数据访问、大数据作为平台的多租户云服务架构 BDaaS 等不同服务模式下，在 Knox、Kerberos 及 DT、SSO、LDAP、Ranger、Sentry 等多种认证和安全服务中，提出了一个大数据联邦访问控制参考模型（FACRM, federation access control reference model）和实施流程，符合面向服务的框架（SOA, service-oriented architecture）。

但以上主要针对 Hadoop 中的数据设置访问控制机制，未考虑密钥的访问控制，不能直接用于密钥访问控制中，同时保证数据和密钥的安全。

1.2　大数据平台的访问控制系统

在现有 Hadoop 访问控制实施方面，HDFS 自带的访问控制系统是 POSIX 权限模型，并支持 POSIX ACL 扩展接口。文件所有者可以依据用户组为文件所有者、文件所有者的同组用户及其他组用户分配读、写、执行等基本操作的权限。

Apache Sentry 是 Hadoop 平台中一个基于角色的访问控制工具，为 Hive、Hive Metastore/HCatalog、Solr、Impala 和存储 Hive 数据的 HDFS 等 Hadoop 服务，提供基于“用户–用户组”“用户组–角色”“角色–权限”对资源访问的统一授权与鉴权。

Apache Ranger 为 Hadoop 集群提供集中式安全框架，支持基于标签的访问控制和策略管理，对 HDFS、Hive、HBase、Knox、Yarn 等组件提供文件级/列级访问控制，提供用户访问的统一审计和安全管理；Ranger KMS 对 Hadoop KMS 进行扩展，为密钥提供数据库安全存储和可视化管理，但不提供与文件统一的细粒度访问控制。

Gupta 等[10]介绍了 Hadoop 的权限管理体系，细数 HDFS、YARN、Hive、Ranger、Knox、Atlas 等 Apache 组件的权限管理作用和参数配置方法，说明 Hadoop 生态系统实例中对服务、数据、应用、基础资源的访问控制框架，并举例示范访问控制框架的应用。但该研究更多地总结了基于 HDP 的组件环境，对 CDH 相关的 Sentry 等组件未进行详细说明。

Ugobame 等[11]将区块链引入大数据访问控制安全生态，借助 Hyperledger Fabric blockchain，实现了区块链基于身份的访问控制商业网络（BIBAC BN, blockchain identity-based access control business network）和区块链基于角色的访问控制（BRBAC, blockchain role-based access control），可以将基于身份或基于角色的操作（如请求、授权、撤权、验证、查看资产等）记录于区块链中，完成身份管理与权限控制，并保护数据隐私。

以上工作已经在 Hadoop 大数据平台中实现了文件的访问控制，但未考虑细粒度访问控制属性，如时态、接入点等环境属性。

1.3 细粒度访问控制模型

除了前述基于角色的访问控制，现有的细粒度访问控制模型包括基于属性的访问控制（ABAC, attribute-based access control）[12]、下一代访问控制（NGAC, next generation access control）[13]、使用控制（UCON, usage control）[14-15]、CoAC[1]等，各具特点。

ABAC 模型采用属性组合描述策略进行赋权，通过策略管理点（PAP）、策略执行点（PEP）、策略决策点（PDP）和策略信息点（PIP）等主体完成访问控制，并采用 XACML 标准[16]实现。属性虽然灵活，但也较为宽泛，在大数据环境中使用时需要具体的定义，而且没有 CoAC 中资源传播链和网络交互图的定义，缺乏对于传播的描述。

NGAC 模型基于图的权限管理模型，通过图来表示主客体之间的继承关系、操作关系，并在此基础上设置策略和权限，易于线性扩展。但在大数据存储环境中，由于大量用户使用，海量文件和密钥作为客体，导致图的信息量难以管理。

UCON 模型在主体、客体、权限的访问控制要素基之上加入了授权、义务和条件 3 个决定性因素，对使用前、使用中和使用后进行使用决策和执行，呈现连续性和可变性的特征。

CoAC 模型涵盖了访问请求实体、广义时态、接入点、访问设备、网络、资源、网络交互图和资源传播链等要素，可有效防止由数据所有权与管理权分离、信息二次/多次转发等带来的安全问题，通过对上述要素的适当调整可描述现有的经典访问控制模型，满足新的信息服务和传播模式的需求，可以借鉴用于解决大数据环境的资源管理问题。

除此之外，还有基于时间的访问控制模型、基于任务的访问控制模型等考虑各类属性的访问控制模型，但属性都相对单一。

1.4 结合密码资源的访问控制机制

随着密钥资源对数据的保护，访问控制的方向增加为：①采用密码协议和计算完成访问控制；②对资源和密钥同时完成访问控制。

CP-ABE[17]、KP-ABE[18]等基于密文策略、属性管理的加密，确实可以同时提供加密和访问控制的功能，但其使用条件不适用于透明加解密的大数据环境，且对加密者的自由度要求很高、管理开销很大。Shafagh 等[19]虽然也将访问控制和密钥统一管理，采用 Dual Hash 树的方式生成密钥，并对密钥进行受控共享，适用场景主要是可穿戴设备等移动应用的时序数据流，提供定长切分保护。但对于大数据环境中的无序独立文件，无法直接提供加密和密钥保护。

2 CKCM 定义及映射

本节首先对面向 HDFS 的密钥资源控制机制（CKCM）进行定义，并呈现从 CoAC 到 CKCM 的映射过程，给出形式化描述。

2.1 CKCM 系统模型

如图 1 所示，本文提出的面向 HDFS 的密钥资源控制机制由 CoAC 的访问控制要素映射至大数据环境，由 HDFS 访问请求实体、网络空间环境、文件系统资源以及密钥资源组成。其中，访问请求实体包括管理员和用户。网络空间环境包括广义时态、接入点、访问设备、网络等。文件系统资源包括 HDFS 中的目录和文件，分别由 HDFS 管理节点 NameNode、数据节点 DataNode 管理和存储。密钥资源为文献[20]中的三层密钥管理体系，即密钥库口令、加密区密钥、文件密钥，由密钥管理系统 KMS 和 HDFS 管理节点 NameNode 存储；文件系统中的加密区与二级密钥绑定，形成透明加解密的加密区，由加密区列表维护。

2.2 从 CoAC 到 CKCM 的映射

CKCM 包括访问请求实体、广义时态、接入点、访问设备、网络、资源、场景等要素的映射。

映射 1 访问请求实体映射。大数据资源访问的发起者包括两类，记为 $q=<u, s>$。其中，用户 u 包括数据访问用户、HDFS 管理员、密钥管理员这 3 类；服务 s 表示通过客户端访问大数据资源的进程，包括 Hive、Solr 等大数据组件的服务，也包括上层应用业务的服务。所有访问请求实体的集合记为 Q。

大数据环境下访问请求实体的通用属性 gAttr 包括用户名（qName）、用户域（qDom）等，以及其他自选添加的属性（qExtGAttr）；安全属性 sAttr 包括登录方式（qPattern）、登录口令（qPwd）、登录证书（qCert）、公私钥对（qKeys）、口令过期时间（qExpTime）等，以及其他自选添加的安全属性（qExtSAttr）。

访问请求实体的通用属性（gAttr）和安全属性（sAttr）可分别表示如下。

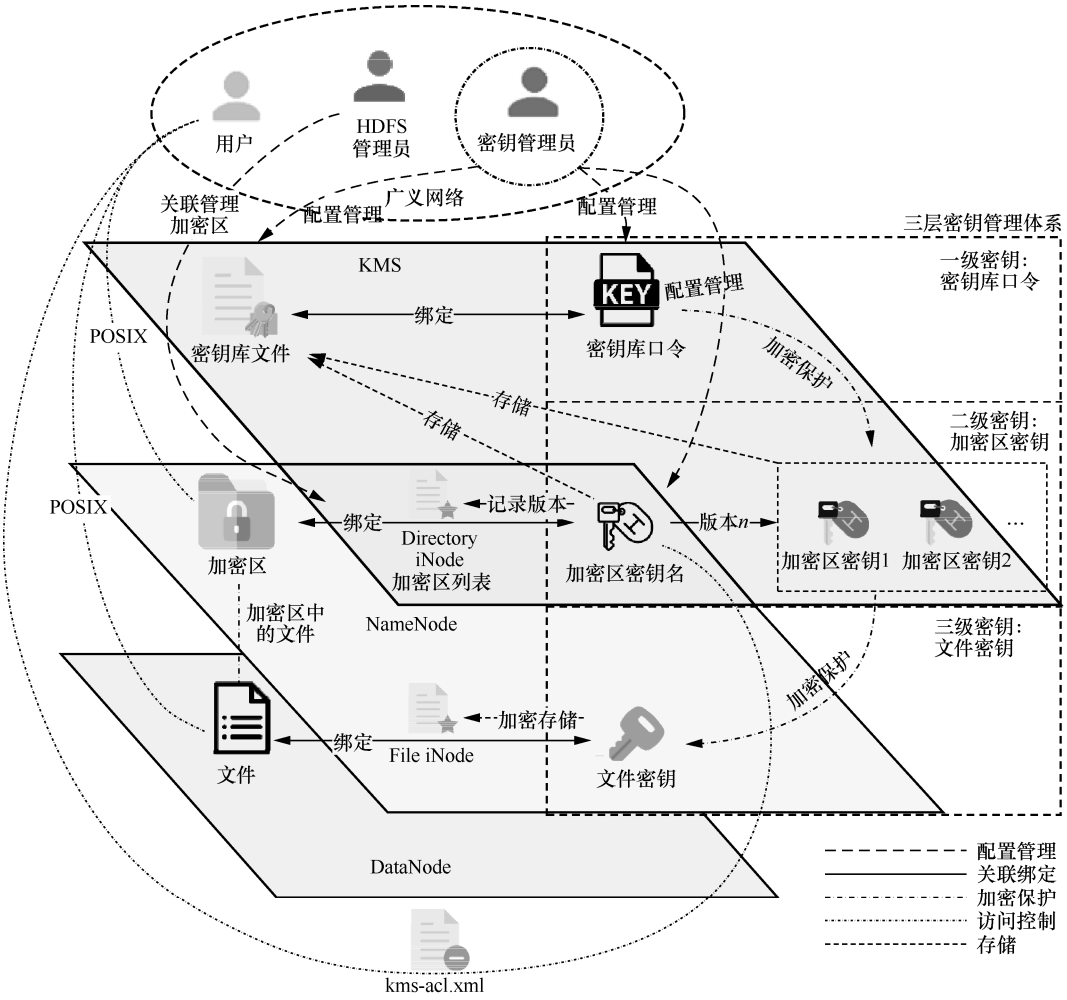

图 1　面向 HDFS 的密钥资源控制机制相关实体

Q.gAttr=<qName, qDom, qExtGAttr, …>

Q.sAttr=< qPattern, qPwd, qCert, qKeys, qExpTime, qExtSAttr, …>

用户域 qDom 主要以 Kerberos 的注册域区分，安全属性 sAttr 也是以 Kerberos 的配置为准。通用属性和安全属性各自支持自选添加属性，由用户或管理员添加和设置，用于后续访问控制策略配置。

映射 2　广义时态映射。与 CoAC 类似，广义时态（GT, general temporal）是指大数据访问请求实体发起访问时，与环境要素的日期时间相关的信息，大数据环境下的时态包含两部分，记为 T。

T={<interval, duration>| interval$\in 2^{T^{IN}}$, duration$\in R^{+}$}

其中，interval$\in 2^{T^{IN}}$ 表示起始日期时间和终止日期时间；duration 表示持续时间，单位可为秒（s）、分钟（min）、小时（h）、天（d）、月（m）、年（y）。

映射 3　接入点映射。接入点（AP, access point）是指大数据访问请求实体首次接入大数据网络中的空间位置或网络标识。大数据环境中以网络标识区分不同访问接入点。

大数据环境接入点的通用属性 gAttr 包括地理位置（aGPS）、MAC 地址（aMac）、IP 地址（aIPv4）、端口（aPort）等；安全属性 sAttr 包括安全传输协议（aSecProt）等。接入点的通用属性（gAttr）和安全属性（sAttr）可分别表示如下。

AP.gAttr=< aGPS, aMac, aIPv4, aPort, …>

AP.sAttr=< aSecProt, …>

其中，安全传输协议 aSecProt 支持 HTTP、HTTP/TLS v1.0 等。

映射 4　访问设备映射。访问设备（DEV, device）是指大数据环境中访问资源时使用的设备，主要包括手机、平板计算机、笔记本计算机、台式

机计算机、服务器等。

访问设备的通用属性（gAttr）包括操作系统（dOS）、处理器（dCPU）、内存（dMem）、硬盘（dDisk）、应用程序（dApp）等；设备的安全属性 sAttr 包括安全等级（dSecLev）、安全软件模块（dSecSMod）、安全硬件模块（dSecHMod）等。

访问设备的通用属性（gAttr）和安全属性（sAttr）可分别表示如下。

DEV.gAttr=< dOS, dCPU, dMem, dDisk, dApp, …>

DEV.sAttr=< dSecLev, dSecSMo, dSecHMod, …>

访问设备 DEV 可用三元组<dID, dev.gAttr, dev.sAttr>表示。其中，dID 为访问设备 ID。

映射 5 网络−大数据平台网络映射。大数据平台（BDP, big data platform）网络是由部署在数据中心的分布式大数据节点组成的网络，是大数据存储、分析、计算的核心载体，包括管理节点（NameNode）、数据节点（DataNode）、计算节点（MapReduce）、认证节点（CertNode）、密钥服务节点（KeyNode）等服务器节点（SN, server node）。BDP 网络可表示为无向联通图 $G_{\mathrm{BDP}}=(V_{\mathrm{BDP}}, E_{\mathrm{BDP}})$，其中，$V_{\mathrm{BDP}}=\{\mathrm{sn}_1,\cdots,\mathrm{sn}_M\}$为图的顶点集合，表示大数据平台的服务器节点集合，$\mathrm{sn}_i$表示第 i 个服务器节点，$1\leqslant i\leqslant M$，$M\geqslant 3$；$E_{\mathrm{BDP}}=\{<\mathrm{sn}_i, \mathrm{sn}_{i+1}>|1\leqslant i\leqslant M, \mathrm{sn}_{M+1}=\mathrm{sn}_1\}$为边集，表示服务器节点之间的传输链路。为了简洁，用 esn 表示大数据平台网络的边。

大数据平台服务器节点 sn 的通用属性 gAttr 包括心跳连接（snHeart）、时间对齐（snTime）、CPU 剩余（snCPU）、内存剩余（snMem）、硬盘剩余（snDisk）、服务状态（snServStatus）、角色实例（snRoleInst）等；大数据平台节点的安全属性 sAttr 包括认证方式（snAuth）等。大数据平台节点的通用属性（gAttr）和安全属性（sAttr）可分别表示如下。

SN.gAttr=< snHeart, snTime, snCPU, snMem, snDisk, snServStatus, snRoleInst, …>

SN.sAttr=< snAuth , …>

大数据平台网络链路 esn 的通用属性 gAttr 包括通信带宽（eWidth）、物理链路层协议（ePhyProt）、网络层协议（eNetProt）、传输层协议（eTranProt）等；大数据平台网络链路 esn 的安全属性 sAttr 包括安全传输协议（eSecProt）、加密算法（eEncType）等。大数据平台网络链路 esn 的通用属性（gAttr）和安全属性（sAttr）可分别表示如下。

ESN.gAttr=< eWidth, ePhyProt, eNetProt, eTranProt, …>

ESN.sAttr=< eSecProt, eEncType, …>

其中，安全传输协议 eSecProt 支持 HTTP、HTTP/TLS v1.0 等；加密算法 eEncType 支持 TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA 等。

映射 6 网络−大数据客户端网络映射。大数据客户端（BDC, big data client）网络是由连接和使用 BDP 网络的客户端组成的网络，大部分客户端之间没有连接，各自只与 BDP 网络有连接。BDC 网络用无向图 $G_{\mathrm{BDC}}=(V_{\mathrm{BDC}}, E_{\mathrm{BDC}})$表示，其中，$V_{\mathrm{BDC}}=\{\mathrm{cn}_1,\cdots,\mathrm{cn}_K\}$为图的顶点集合，表示大数据平台的客户端节点集合，$\mathrm{cn}_i$表示第 i 个客户端节点，$1\leqslant i\leqslant K$，$K\geqslant 1$；$E_{\mathrm{BDC}}=\{<\mathrm{sn}_i, \mathrm{sn}_{i+1}>|1\leqslant i\leqslant M, \mathrm{sn}_{M+1}=\mathrm{sn}_1\}$为边集，表示客户端节点之间的传输链路。BDC 网络中节点和链路的属性类型，与 BDP 网络中节点和链路的属性类型相同，属性值存在差异，此处不再重复属性类型。

映射 7 网络映射。大数据网络（BDN, big data network）是信息传播的载体，是所有信息传播通道的集合。BDN 可以用无向图 $G_{\mathrm{BDN}}=(V_{\mathrm{BDN}}, E_{\mathrm{BDN}})$表示，该网络的顶点包括 BDP 网络顶点 V_{BDN} 和 BDC 网络顶点 V_{BDC}，边 E_{BDN} 由 E_{BDP} 和 E_{BDC} 组成，即 $V_{\mathrm{BDN}}=V_{\mathrm{BDP}}\cup V_{\mathrm{BDC}}$，$E_{\mathrm{BDN}}=E_{\mathrm{BDP}}\cup E_{\mathrm{BDC}}\cup\{<\mathrm{cn}_i, \mathrm{sn}_j>, 1\leqslant i\leqslant M, 1\leqslant j\leqslant K,$ 大数据客户端节点 cn_i 与大数据平台服务器节点 sn_j 可连接$\}$

大数据网络中的通用属性 BDN.gAttr 和安全属性 BDN.sAttr 包括顶点属性和边属性，顶点属性为图 G_{BDP} 和图 G_{BDC} 所有顶点属性的并集，边属性为图 G_{BDP} 和图 G_{BDC} 中所有边属性的并集。

映射 8 资源−HDFS 文件系统映射。HDFS 文件系统（FS, file system）资源包括目录（Directory，记为 DIR）和文件（File，记为 FILE）两部分，可用二元组<inodeID, path>表示，其中，inodeID 为文件系统资源对应的唯一标识 inode 号，path 为目录或文件的路径，路径可变、inode 号不可变。目录信息由大数据平台网络的管理节点管理，文件数据由大数据平台网络的数据节点存储。

目录的通用属性包括目录名（dirName）、目录最后一次修改时间（dirModTime）、目录下子节点个数最大值（dirNsQuota）、目录下所有文件占用存储空间最大值（dirDsQuota）等。目录的安全属性

包括目录权限（dirPerm）、扩展属性（dirXAttrs）等。目录的基本权限为 POSIX ACL 的 rwx（read-write-execute）形式。目录的通用属性（gAttr）和安全属性（sAttr）可分别表示如下。

DIR.gAttr=< dirName, dirModTime, dirNsQuota, dirDsQuota, …>

DIR.sAttr=< dirPerm, dirXAttrs,…>

文件的通用属性包括文件名（fileName）、文件最后一次访问时间（fileAccessTime）、文件最后一次修改时间（fileModTime）、文件副本数（fileReplica）等。文件的安全属性包括文件权限（filePerm）、扩展属性（fileXAttrs）等。文件权限和目录权限的基本标准相同，为 POSIX ACL 的 rwx 形式。文件的通用属性（gAttr）和安全属性（sAttr）可分别表示如下。

FILE.gAttr=< fileName, fileAccessTime, fileModTime, fileReplica, …>

FILE.sAttr=< filePerm, fileXAttrs, …>

定义 1　HDFS 文件系统层次结构（FSH, file system hierarchy）。HDFS 文件系统层次结构 FSH⊆DIR×FILE 是 HDFS 文件系统中的偏序关系。

从根目录（/）开始，目录（DIR）与文件（FILE）构成偏序关系 DIR≥FILE，当且仅当该文件位于该目录中。

映射 9　资源−KMS 密钥体系映射。KMS 密钥体系是指大数据文件的密钥管理体系，包括密钥库口令（OPwd）、加密区密钥（OEZK）、文件密钥（OFK）这 3 层，可分别称为一级密钥、二级密钥和三级密钥。KMS 密钥资源（rKMS）可用三元组<kid, kgAttr, ksAttr>表示，其中，kid 表示 rKMS 的体系层，kgAttr 和 ksAttr 分别表示密钥的通用属性和安全属性。

密钥库口令资源的通用属性 OPwd.kgAttr 包括拥有者（pOwner）、初始化时间（pIniTime）、口令存储位置（pStoreLoc）等。安全属性 OPwd.ksAttr 包括口令长度（pLength）、默认口令（pDefault）、加密类型（pEncType）等，其中，口令长度 pLength 为 128 位或 256 位；默认口令 pDefault 为字符串“Password”的摘要值；加密类型 pEncType 包括 DES、SM4 等。密钥库口令资源 OPwd 的通用属性（kgAttr）和安全属性（ksAttr）可分别表示如下。

OPwd.kgAttr=< pOwner, pIniTime, pStoreLoc, …>

OPwd.ksAttr=< pLength, pDefault, pEncType, …>

加密区密钥资源的通用属性 OEZK.kgAttr 包括密钥名（zkKeyName）、创建时间（zkCreateTime）、密钥描述信息（zkDescrip）、密钥属性对（zkAttrPairs）等。加密区密钥资源的安全属性 OEZK.ksAttr 包括加密算法（zkCipher）、密钥长度（zkLength）等。其中，加密算法 zkCipher 有 AES/CTR/NoPadding、SM4/CTR/NoPadding 等算法与模式组合，用于指定所绑定加密区中文件加密算法；密钥长度为 128 位、256 位等。加密区密钥资源 OEZK 的通用属性（kgAttr）和安全属性（ksAttr）可分别表示如下。

OEZK.kgAttr = <zkKeyName, zkCreateTime, zkDescrip, zkAttrPairs, …>

OEZK.ksAttr=< zkLength, zkCipher, …>

文件密钥资源的通用属性 OFK.kgAttr 包括一次一密文件路径（fkPath）等。文件密钥资源的安全属性 OFK.ksAttr 包括密钥校验值（fkCheckSum）等。文件密钥资源 OFK 的通用属性（kgAttr）和安全属性（ksAttr）可分别表示如下。

OFK.kgAttr=<fkPath, …>

OFK.ksAttr=<fkCheckSum, …>

定义 2　密钥偏序关系。KMS 密钥资源层次结构（KMSCH, key management system hierarchy）KMSCH⊆ OPwd×OEZK×OFK 是 KMS 密钥资源中的偏序关系，由密钥库口令（OPwd）、加密区密钥（OEZK）、文件密钥（OFK）组成。

密钥库口令（OPwd）与所有加密区密钥（OEZK）构成偏序关系 OPwd≥OEZK；加密区密钥（OEZK）与文件密钥（OFK）构成偏序关系 OEZK≥OFK，当且仅当文件密钥所保护的文件位于加密区密钥所保护的加密区中。

映射 10　资源−加密区管理列表映射。加密区管理（EZM, encryption zone manager）模块是对目录与加密区密钥的绑定关系管理，维护加密区管理列表，列表中的关联关系可用目录名和加密区密钥名组成的二元组<dirName, zkKeyName>表示。用户在访问加密区中的目录和文件时，需通过加密区管理模块获取密钥信息。

映射 11　资源映射。大数据资源（BDR, big data resource）是大数据环境中访问请求实体访问的对象，是所有访问对象的集合，包括 HDFS 文件系统、KMS 密钥体系、加密区管理列表等。即 BDR=HDFS∪KMS∪EZM。

资源的通用属性 BDR.gAttr 和安全属性

BDR.sAttr，分别为 HDFS 文件系统（目录 DIR 和文件 FILE）、KMS 密钥体系（密钥库口令 OPwd、加密区密钥 OEZK 和文件密钥 OFK）、加密区管理列表 EZM 中所有通用属性和安全属性的并集。

定义 3　操作（Operation）。大数据环境下的操作是指访问请求实体对资源的访问动作，通过动作的连续进行而实现大量的系统功能。根据主要访问的目标资源，操作可以分为密钥管理操作（KMO, key management operation）、加密区操作（EZO, encryption zone operation）、HDFS 操作（DFSO, distributed file system operation）等。

实体的一个操作可能影响多个资源，操作涉及的客体如表 1 所示。其中，“√”表示一定存在影响，如密钥管理操作对各级密钥存在影响、加密区操作对相关绑定实体存在影响、HDFS 操作对文件和目录存在影响；“○”表示关联绑定加密区后存在影响，如更新绑定了加密区的密钥，将影响该加密区中此后上传的最新密钥版本等，对于未绑定加密区的密钥，密钥管理操作不影响加密区和文件等资源。

映射 12　场景（SC, scene）。场景指大数据访问请求主体 q 启动会话 s 访问大数据资源 r 获得权限 p 时所需要的要素条件，要素包括广义时态、接入点、设备、大数据网络及其属性，以及访问请求实体 q 和资源 r 的属性，记为 sc，由五元组(t, l, d, bdn, attr)表示。其中，$t \in T$, $l \in L$, $d \in$ Dev, bdn ∈ BDN, attr ∈ {gAttr ∪ sAttr}。

与 CoAC 模型中场景的不同之处在于，CKCM 机制的场景除了四要素外，还包括所有要素的属性。场景具备偏序关系，可包含子场景。

2.3　CKCM 实施/生效机制、核心资源控制函数

CKCM 机制的核心问题是授权和鉴权。为高效、准确地授予和撤销访问权限，本文借鉴 CoAC 中对权限的管理方式，通过场景和属性来设置大数据环境下的约束策略，分配和撤销用户对资源的访问权限（P, privilege）。场景由广义时态、接入点、设备、大数据网络这 4 个要素构成，属性则是访问请求实体、场景四要素、资源属性的并集。

在大数据资源开放访问之前，预先设置约束策略，约定具备何种属性的主体在何种场景下对具备何种属性的何种资源可执行何种操作（完成“场景–权限”分配）；未分配权限的场景默认拒绝访问。当大数据访问请求实体 Q 对资源 BDR 发起访问请求时，通过分配会话（S, session）权限执行点给出访问请求实体的场景和属性信息（完成“用户–场景”信息收集），权限决策点基于“用户–场景”信息和预分配的“场景–权限”进行场景匹配，根据场景匹配到的权限，判定 Q 是否有权对 BDR 执行操作。面向 HDFS 的密钥资源控制机制如图 2 所示。

相关访问控制函数如下。

表 1　　**操作涉及的客体**

操作类型	操作	涉及修改和访问的资源					
		KMS			HDFS		EZM
		一级密钥	二级密钥	三级密钥	NN	DN	
密钥管理操作（KMO）	创建二级密钥	√	√	—	—	—	—
	更新二级密钥	√	√	—	○	—	○
	删除二级密钥	—	√	—	○	○	○
	三级密钥缓冲池管理	√	√	√	—	—	—
	查二级密钥名	—	√	—	—	—	—
	查二级密钥内容	√	√	—	—	—	—
加密区操作（EZO）	绑定加密区	—	√		√	—	√
	向加密区上传/下载文件	√	√	○	√	√	√
HDFS 操作（DFSO）	初始化文件系统	—	—	—	√	√	—
	创建空文件夹	—	—	—	√	—	—
	非加密区文件上传/下载	—	—	—	√	√	—
	修改文件默认权限	—	—	—	√	—	—

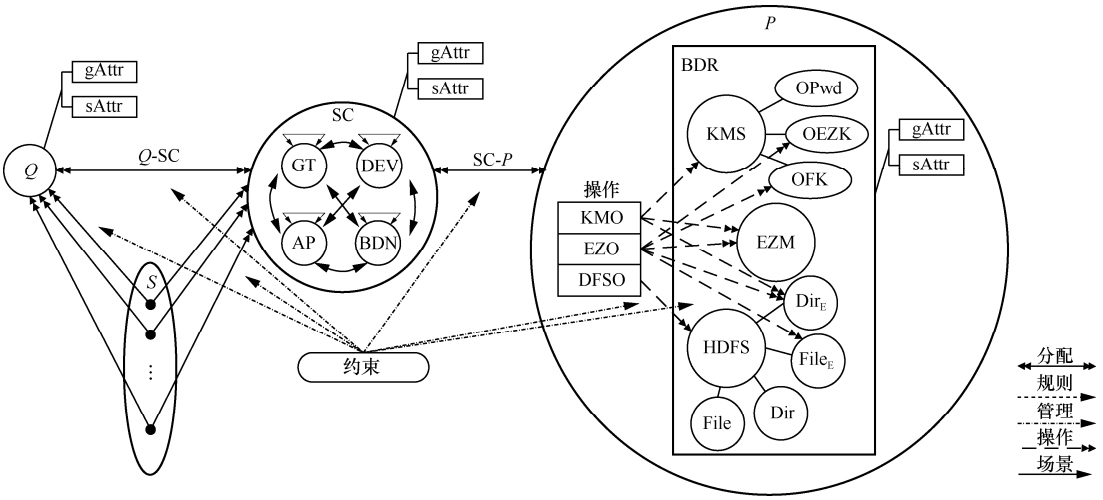

图 2　面向 HDFS 的密钥资源控制机制

1) attrValueCollect(Q.gAttr, AP.gAttr, DEV.gAttr, BDN.gAttr, BDR.gAttr, Q.sAttr, AP.sAttr, DEV.sAttr, BDN.sAttr, BDR.sAttr)→{gAttr_value∪sAttr_value}为属性值采集函数，用于确定当前访问请求中的相关属性值，形成访问请求场景，备用于场景构建和匹配。

2) sceneConstruct(gAttr_value∪sAttr_value)→{Q.SCENE}为场景构建函数，用于将访问请求中的属性及属性值进行归集，形成场景实例，备用于匹配策略中的场景。

3) sceneMatch(Q.SCENE⊆SCENE)→{true,false}为场景匹配函数，将访问请求的实例场景与策略中配置的场景进行匹配，确实实例场景是否为配置场景的子集，其中，SCENE 表示管理员已配置的场景集合。

4) scenePermActivate(SCENE,PERM)→{true,false}为场景−权限激活函数，用于激活给定场景所具备的权限，其中，true 表示激活成功，false 表示激活失败。

5) scenePermRevoke(SCENE,PERM)→{true,false}为场景−权限撤销函数，用于撤销指定场景中已具备、可执行的权限。

3　CKCM 管理模型

大数据环境具有数据量大、分布式存储的特点，多用户跨网交叉访问，数据跨服务流动，使大数据环境下的海量文件和密钥控制复杂，需要设计访问控制的管理模型和管理函数，确保面向 HDFS 的密钥资源控制机制安全高效地运行。本文将对管理过程进行分析，并采用 Z 语言规范描述管理函数。

3.1　面向 HDFS 的密钥资源控制机制管理模型

在大数据环境中，管理员通过服务管理系统和策略管理系统，在给定的时间段，通过特定设备、网络对访问请求实体访问特定资源的属性、场景、权限进行配置、撤销和更新管理。图 3 给出了面向 HDFS 的密钥资源控制机制管理模型。

定义 4　管理场景（ADSC, administration scene）。该场景由四元组<adminT, adminAP, adminDEV, adminBDN>表示，意为管理员在 adminT 时间，通过 adminDEV 设备，从接入点 adminAP，经由 adminBDN 进行管理的场景。

大数据环境的管理员包括超级管理员、HDFS 管理员、KMS 管理员、属性管理员、策略管理员。各类管理者通过管理场景对访问场景、访问权限进行管理，其管理过程涉及的管理操作流程如下。

①超级管理员为其他管理员分配、撤销管理场景，并维护管理场景对应的权限。②HDFS 管理员通过管理场景，维护 HDFS 文件系统管理和加密区服务管理。③KMS 管理员通过管理场景，维护密钥系统管理。④属性管理员通过管理场景，创建、删除和更新访问请求实体、资源及场景四要素的属性元数据，并配置属性值。⑤策略管理员通过管理场景，汇聚属性形成场景，并对场景进行授权和撤销。⑥访问请求实体发起访问时，管理模型认证身份并分配会话，根据其访问的时间、设备、接入点、大数据网络及各属性形成场景，判定场景是否满足策略中该访问请求实体所具备权限的场景，若是则激活权限，反之则拒绝访问。

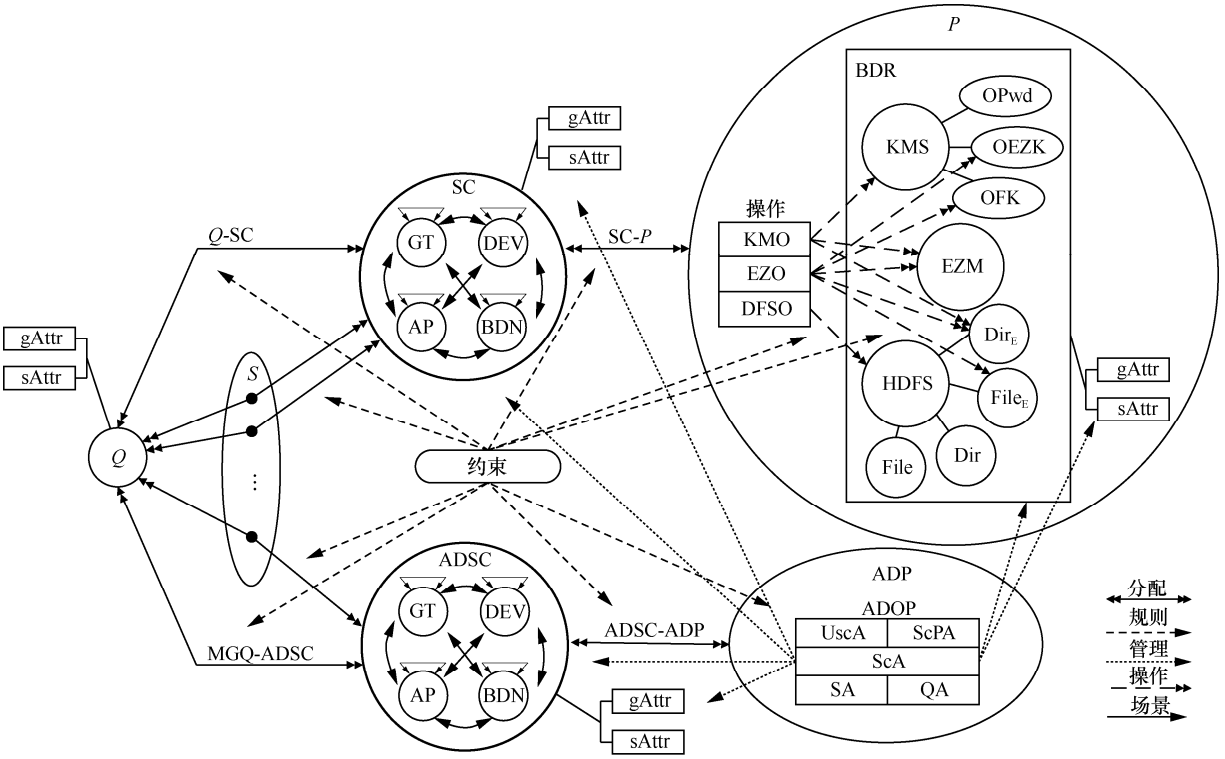

图 3　面向 HDFS 的密钥资源控制机制管理模型

根据上述内容，管理对象包括：① 访问请求实体的身份及属性，场景中的元素及属性，资源的操作及属性，策略中的场景及权限；② 所有元素的属性管理，访问请求实体的场景获取，给当前访问请求实体分配会话，给当前访问请求实体所处场景分配权限。

基于管理场景，本文的管理模型为文献[1]的实例化，因此不再给出管理模型组件的定义。同时，本文的管理员权限分配与撤销，与文献[21]一致，这里不再赘述。

3.2　面向 HDFS 的密钥资源控制机制管理函数

大数据环境下的访问控制管理模型，主要包括 UScA（用户−属性场景管理）、ScPA（属性场景−权限管理）、ScA（属性场景管理）这 3 个主要管理子模型，同时还需包括访问请求主体身份管理 QA、会话管理 SA 等支撑模型。

CKCM 管理函数的语义复杂，采用自然语言描述易出现歧义，需要使用形式化语言描述，以确保描述的正确性。Z 形式化描述语言（简称“Z 语言”）[16]是一种书写软件规格说明的方法，采用形式化的方法描述和验证软件系统的需求、功能、规格等，为设计、编程和测试提供理论依据，从而确保软件的正确性、可靠性和精细化描述。因此，为确保语义准确，本节通过 Z 语言给出上述 5 类子模型的函数描述，提供理论模型。

3.2.1　管理子模型

根据管理流程，3 个管理子模型依次描述如下。

1) 属性场景管理

属性场景管理（ScA, scene administration）是对各要素属性元数据、场景要素的管理，是权限管理和用户场景管理的基础，包括 createMetaAttr、deleteMetaAttr 、 selectFunction 、 createScene 、modScene、deleteScene 等函数，如表 2 所示，功能分别为创建属性元数据、删除属性元数据、选取场景描述函数、创建场景、修改场景、删除场景等。

2) 属性场景−权限管理

属性场景−权限管理（ScPA, scene and privilege administration）是指为已创建的场景授予和撤销权限，包括 assignScPerm 和 revokeScPerm，分别对应授予场景权限、撤销场景权限的功能，如表 3 所示。本文模型中，授予权限即允许访问，撤销权限即拒绝访问，不具备场景和权限时，默认为拒绝。属性和场景不具备继承关系。场景出现重叠时，授权均为允许，重叠场景的组合结果也是允许，因此不存在权限冲突的问题。

3) 用户−属性场景管理

用户−属性场景管理（UScA, user and scene ad-

表 2　　**属性场景管理类**

函数	描述
创建属性元数据 createMetaAttr	添加元素的属性元数据，需选择数据类型、管理对象，填写属性名、属性描述（可选），若添加成功，返回 true；否则返回 false createMetaAttr(METAATTR?, p?:AttrName, ds?:DataSturcture, elt?:Element, des?:Description, result!:Boolean)◁ if p ∉ elt.MetaAttr then meta=<p, ds, elt, des>, METAATTR'= METAATTR ∪ { meta }, result=true else result=false▷
删除属性元数据 deleteMetaAttr	删除元素的属性元数据及已配置的属性值，若删除成功，返回 true；否则返回 false deleteMetaAttr(METAATTR?, p?:AttrName, result!:Boolean)◁ if p↣meta∈METAATTR, then METAATTR'=METAATTR∖{p↣meta}, result=true else result=false▷
选取场景描述函数 selectFunction	选择场景描述函数，如各数据类型元素的对比和计算函数，根据属性的数据类型输出 selectFunction(elt?:Element, func!:Function)◁ getFunction(getDataStructure(elt))→func▷
创建场景 createScene	选取已有要素、属性和函数，搭建场景，若创建成功，返回 true；否则返回 false createScene(SCENE?, q?:Quantity, rsc?Resource, (t, ap, dev, bdn)?:Name, result!:Boolean)◁ sc=selectFunction((t, ap, dev, bdn), getAttr(q, rsc)) if sc ∉ SCENE then SCENE'=SCENE∪{sc}, result=true else result=false▷
修改场景 modScene	修改已有场景中的要素、属性和函数，若修改成功，返回 true；否则返回 false modScene(SCENE?, sca?,scb?:Scene, result!:Boolean)◁ if sca∈SCENE and scb ∉ SCENE then SCENE'=SCENE∪{scb}∖{sca}, result=true else result=false▷
删除场景 deleteScene	删除已有场景，若删除成功，返回 true；否则返回 false deleteScene(SCENE?, sc?:Scene, result!:Boolean)◁ if sc∈SCENE then SCENE'=SCENE∖{sc}, result=true else result=false▷

表 3　　**属性场景–权限管理类**

函数	描述
授予场景权限 assignScPerm	为已创建的场景分配权限，若分配成功，返回 true；否则返回 false assignScPerm(PERM?, sc?:Scene, p?:Perm, result!: Boolean)◁ if sc∈SCENE and p ∉ PERM then PERM'=PERM∪{p}, result=true else result=false▷
撤销场景权限 revokeScPerm	撤销场景所具有的权限，若撤销成功，返回 true；否则返回 false assignScPerm(PERM?, p?:Perm, result!: Boolean)◁ if p∈PERM then PERM'=PERM∖{p}, result=true else result=false▷

ministration）是指管理用户的属性值，在用户发起访问请求时，获取并分配场景。函数包括 modAttr、getAttr 、 getAttrValue 、 verifyTime 、 verifyAP 、verifyDevice、verifyBDNetwork 等，功能分别为添加/修改属性值、选取/查询属性、查询属性值、验证时间、验证接入点、验证设备、验证网络，如表 4 所示。

3.2.2　管理支撑模型

管理支撑模型为管理场景和管理权限的运行实施提供基础，是面向 HDFS 的密钥资源控制机制的必要组成。

1)　访问请求实体身份管理

访问请求实体身份管理（QA, quantity administration）意在管理用户身份，提供身份认证，为会话管理、用户–属性场景管理提供支撑。函数包括 QuantitySignUp 、 QuantityLogOff 、updateAuthentication、CheckId、getQuantityList，

表 4 用户–属性场景管理类

函数	描述
添加/修改属性值 modAttr	添加或修改元素的属性值，若添加成功，返回 true；否则返回 false modAttr(ATTRVALUE?, p?:AttrName, elt?:Element, vlu?:Value, result!:Boolean)◁ if (p∈elt.MetaAttr) and (vlu∈allowedValue) then attrValue=<p,elt, vlu>, ATTRVALUE'= ATTRVALUE ∪ { attrValue }, result=true else result=false▷
选取/查询属性 getAttr	选取/查询各元素的属性，用于构建场景，若查询成功，返回属性集合；否则返回 null getAttr(ATTR?, elt?:Element, result!: AttrSet)◁ if checkId(elt) then result=elt↣ATTR else result=null▷
查询属性值 getAttrValue	查询访问请求实体的所有属性值，用于匹配场景，若查询成功，返回属性值集合；否则返回 null getAttrValue(ATTRVALUE?, q?:Quantity, result!: AttrValueSet)◁ if checkId(q) then result=q↣ATTRVALUE else result=null▷
验证时间 verifyTime	验证时间在场景的允许范围内，若通过验证，返回 true；否则返回 false verifyTime(sc?:SCENE, t?:Name, result!:Boolean)◁ if t∈sc.t then result=true else result=false▷
验证接入点 verifyAP	验证接入点在场景的允许范围内，若通过验证，返回 true；否则返回 false verifyAP(sc?:SCENE, ap?:Name, result!:Boolean)◁ if ap∈sc.ap then result=true else result=false▷
验证设备 verifyDevice	验证设备在场景的允许范围内，若通过验证，返回 true；否则返回 false verifyDevice(sc?:SCENE, dev?:Name, result!:Boolean)◁ if dev∈sc.dev then result=true else result=false▷
验证网络 verifyBDNetwork	验证设备在场景的允许范围内，若通过验证，返回 true；否则返回 false verifyBDNetwork(sc?:SCENE, bdn?:Name, result!:Boolean)◁ if bdn∈sc.bdn then result=true else result=false▷

功能分别为用户实体注册，用户实体删除/注销，口令/证书的初始化、更新、撤销、口令/证书认证（用于会话建立之前）、用户列表查询（用于属性场景管理和权限管理）等，如表 5 所示。

2) 会话管理

会话管理（SA, session administration）是指为每一次通过认证的访问请求创建或关闭会话，用于访问请求实体身份认证之后、激活场景权限之前。函数包括 assignQSession、assignScSession、closeSession 等，功能分别是为通过认证的用户分配会话、为会话分配场景、关闭会话等，如表 6 所示。

3.3 CKCM 模型分析

下面，分析 CKCM 及其管理模型如何满足引言中提出的安全需求。

1) 支持密钥资源控制。CKCM 将三层密钥管理体系纳入资源管理范畴，在会话、属性、场景、权限的管理中，均支持统一的管理力度和控制效果。

2) 支持操作语义归一化描述。该机制通过 Z 语言形式化描述管理函数和管理操作，支持 HDFS、KMS、EZM 的资源控制统一描述和鉴权，因而该机制支持操作语义归一化描述。

3) 支持细粒度访问控制。该机制通过会话、属性、场景、权限的管理，支持对于访问主体、资源、时间、设备、网络 IP、操作、资源等要素和属性的控制。采用这些要素可以实现大数据存储系统的细粒度访问控制。

本文所提资源控制机制除了支持这 3 个核心需

表 5　　实体身份与认证管理类

函数	描述
用户实体注册 QuantitySignUp	注册未重名的用户实体，并更新，若注册成功，返回 true；否则返回 false QuantitySignUp(Quantity?, Password?, Certification?, q?:Name, result!:Boolean)◁ if q ∉ Quantity then Quantity'= Quantity∪ {q}, Password'=Password∪ {q↣pwd}, Certification'=Certification∪ {q↣cert} result=true else result=false▷
用户实体删除/注销 QuantityLogOff	删除用户实体，若删除成功，返回 true；否则返回 false QuantityLogOff(Quantity?, Password?, Certification?, q?:Name, result!:Boolean)◁ if q∈Quantity then Quantity'= Quantity∖{q}, Password'=Password∖{q↣pwd}, Certification'=Certification∖{q↣cert} result=true else result=false▷
口令/证书的初始化、更新、撤销 updateAuthentication	更新用户实体的口令和证书，若更新成功，返回 true；否则返回 false updateAuthentication(Quantity?, q?:Name, result!:Boolean)◁ if q∈Quantity then Password'=Password∪ {q↣NewPwd}, Certification'=Certification∪ {q↣NewCert}, result=true else result=false ▷
口令/证书认证 CheckId	验证用户身份与口令/证书，若验证成功，返回 true；否则返回 false CheckId(Quantity?, q?:Name, pwd?:Password, cert?:Certification, result!:Boolean)◁ if (q∈Quantity) and ((pwd⊆(q↣password)) or (cert⊆(q↣certification))) then result=true else result=false▷
用户列表查询 getQuantityList	返回当前用户列表 getQuantityList(Quantity?, list!:QuantityList)◁ getId(Quantity)→list▷

表 6　　会话管理类

函数	描述
为通过认证的用户分配会话 assignQSession	认证用户身份并分配会话，若分配成功，返回 true；否则返回 false assignSession(QSession?, q?:Name, result!:Boolean)◁ if checkId(q) then QSesson'=QSession∪ {<q, s>}, result=true else result=false▷
为会话分配场景 assignScSession	获取场景，并分配给会话，若分配成功，返回 true；否则返回 false assignScSession(ScSession?, q?,s?:Name, result!:Boolean)◁ if createScene(q)→sc then ScSession'=ScSession∪ {<s, sc>}, result=true else result=false▷
关闭会话 closeSession	关闭会话，若关闭成功，返回 true；否则返回 false closeSession(QSession?, ScSession?, q?,s?:Name, sc?Scene, result!:Boolean)◁ if (<q, s>∈QSession) and (<s, sc>∈ScSession) then QSesson'=QSession∖{<q, s>}, ScSession'=ScSession∖{<s, sc>} else result=false ▷

求外，还支持属性自定义和策略自定义。通过属性场景管理类的函数，可以自定义要素属性和属性值。通过属性场景−权限管理类的函数，可以根据已有属性、属性值和场景，设置资源控制策略。对大数据存储系统的文件与密钥提供统一、灵活的资源管理，同时对使用 HDFS 文件系统的上层 Hadoop 服务提供统一的数据控制，实现文件与密钥统一访问控制管理。

4　实验分析

4.1　系统实现

本文在 Hadoop 环境中实施面向 HDFS 的密钥资源控制 CKCM 模型和管理模型，基于 XACML[18] 策略表示方式，设置策略执行点（PEP, policy enforcement point）、上下文处理（context handle）、策略判定点（PDP, policy determination point）、策略信

息点（PIP, policy information point）、策略管理点（PAP, policy administration point），实现 HDFS、KMS 和 EZM 中属性和策略的统一管理与评估，如图 4 所示。CKCM 系统实施不改变 HDFS 接口与架构。

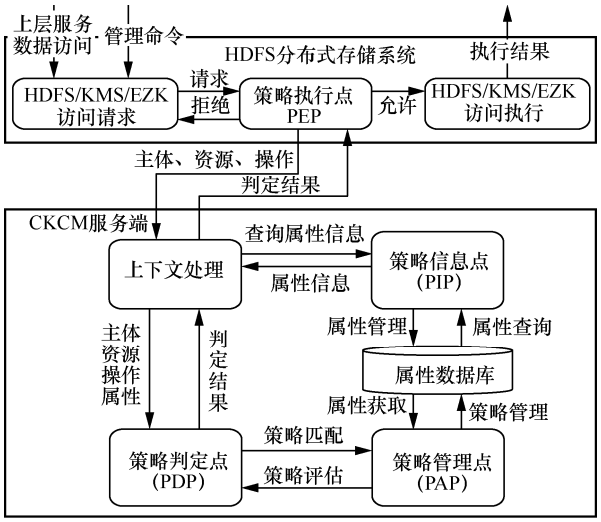

图 4 面向 HDFS 的密钥资源管理 CKCM 模型实施

4.2 性能分析

经过系统测试，表 7 给出了系统确定资源控制策略判定时空消耗随访问请求数量的变化趋势。本节分别测试 8 种使用场景下的系统响应时间和内存占用大小，模拟总访问请求连接数为 2 000、5 000、10 000、15 000 个，并发访问请求连接数占比分别为 50%和 75%的情况下，系统对于每个连接的平均响应时长，以及各场景下的服务器内存占用。

从表 7 中可以看出，本文系统的策略判定时间开销随着并发访问请求连接数的增加而增加，近似呈线性增长趋势。当总访问请求连接数达到 15 000 个、并发访问请求连接数达到 12 000 个时，策略判定的平均响应时间为 1.6 ms，对于文件与密钥资源控制而言，该时间在可接受范围内。

5 结束语

大数据环境下服务众多、数据海量、密钥倍增，本文针对大数据环境对密钥资源的细粒度访问控制、操作语义归一化描述等需求，从密钥资源控制的要素及属性出发，通过映射面向网络空间的访问控制模型，提出了面向 HDFS 的密钥资源控制机制；同时，给出了面向 HDFS 密钥资源控制机制的管理模型，并用 Z 语言形式化地描述了管理模型中的管理函数和管理方法；最后，基于 XACML 实现 CKCM 系统，实现了文件与密钥统一访问控制管理。

参考文献：

[1] 李凤华, 王彦超, 殷丽华, 等. 面向网络空间的访问控制模型[J]. 通信学报, 2016, 37(5): 9-20.
LI F H, WANG Y C, YIN L H, et al. Novel cyberspace-oriented access control model[J]. Journal on Communications, 2016, 37(5): 9-20.

[2] COLOMBO P, FERRARI E. Access control in the era of big data: state of the art and research directions[C]//Proceedings of the 23rd ACM on Symposium on Access Control Models and Technologies. New York: ACM Press, 2018: 185-192.

[3] ULUSOY H, COLOMBO P, FERRARI E, et al. GuardMR: fine-grained security policy enforcement for MapReduce systems[C]//Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security. New York: ACM Press, 2015: 285-296.

[4] ULUSOY H, KANTARCIOGLU M, PATTUK E, et al. Vigiles: fine-grained access control for MapReduce systems[C]//Proceedings of 2014 IEEE International Congress on Big Data. Piscataway: IEEE Press, 2014: 40-47.

[5] GUPTA M, PATWA F, SANDHU R. POSTER: access control model for the hadoop ecosystem[C]//Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies. New York: ACM Press, 2017: 125-127.

[6] GUPTA M, PATWA F, SANDHU R. Object-tagged RBAC model for the Hadoop ecosystem[C]//Data and Applications Security and Privacy (DBSec). Berlin: Springer, 2017: 63-81.

[7] SANDHU R S, COYNE E J, FEINSTEIN H L, et al. Role-based

表 7 资源控制策略判定时空消耗随访问请求数量的变化趋势

总访问请求连接数/个	并发访问请求连接数占比	平均响应时间/ms	内存占用/GB
2 000	50%	0.237 4	0.5
2 000	75%	0.264 2	0.6
5 000	50%	0.309	0.7
5 000	75%	0.326 3	0.8
10 000	50%	0.555 5	0.8
10 000	75%	0.784 9	0.9
15 000	50%	0.967 8	1.3
15 000	75%	1.689 9	1.4

access control models[J]. Computer, 1996, 29(2): 38-47.
[8] GUPTA M, PATWA F, SANDHU R. An attribute-based access control model for secure big data processing in Hadoop ecosystem[C]//Proceedings of the 3rd ACM Workshop on Attribute-Based Access Control. New York: ACM Press, 2018: 13-24.
[9] AWAYSHEH F M, ALAZAB M, GUPTA M, et al. Next-generation big data federation access control: a reference model[J]. Future Generation Computer Systems, 2020, 108: 726-741.
[10] GUPTA M, PATWA F, BENSON J, et al. Multi-layer authorization framework for a representative Hadoop ecosystem deployment[C]//Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies. New York: ACM Press, 2017: 183-190.
[11] UGOBAME U U, SCHNEIDER K A, HOSSEINZADEH K S, et al. Blockchain access control ecosystem for big data security[C]//Proceedings of 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data. Piscataway: IEEE Press, 2018: 1373-1378.
[12] HU V C, FERRAIOLO D, KUHN R, et al. Guide to attribute based access control (ABAC) definition and considerations[R]. 2014.
[13] MELL P, SHOOK J, HARANG R, et al. Linear time algorithms to restrict insider access using multi-policy access control systems[J]. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 2017, 8(1): 4-25.
[14] SANDHU R, PARK J. Usage control: a vision for next generation access control[C]//Computer Network Security. Berlin: Springer, 2003: 17-31.
[15] BALDI G, DIAZ-TELLEZ Y, DIMITRAKOS T, et al. Session-dependent usage control for big data[J]. Journal of Internet Services and Information Security, 2020, 10(3): 76-92.
[16] OASIS Open. OASIS eXtensible access control markup language (XACML) TC version 3.0[EB]. 2013.
[17] PREMKAMAL P K, PASUPULETI S K, ALPHONSE P J A. A new verifiable outsourced ciphertext-policy attribute based encryption for big data privacy and access control in cloud[J]. Journal of Ambient Intelligence and Humanized Computing, 2019, 10(7): 2693-2707.
[18] KAPIL G, AGRAWAL A, ATTAALLAH A, et al. Attribute based honey encryption algorithm for securing big data: Hadoop distributed file system perspective[J]. PeerJ Computer Science, 2020, 6: e259.
[19] SHAFAGH H, BURKHALTER L, RATNASAMY S, et al. Droplet: decentralized authorization and access control for encrypted data streams[C]//Proceedings of the 29th USENIX Conference on Security Symposium. Berkeley: USENIX Association, 2020: 2469-2486.
[20] 金伟, 余铭洁, 李凤华, 等. 支持高并发的Hadoop高性能加密方法研究[J]. 通信学报, 2019, 40(12): 29-40.
JIN W, YU M J, LI F H, et al. High-performance and high-concurrency encryption scheme for Hadoop platform[J]. Journal on Communications, 2019, 40(12): 29-40.
[21] 李凤华, 陈天柱, 王震, 等. 复杂网络环境下跨网访问控制机制[J]. 通信学报, 2018, 39(2): 1-10.
LI F H, CHEN T Z, WANG Z, et al. Cross-network access control mechanism for complex network environment[J]. Journal on Communications, 2018, 39(2): 1-10.
[22] DÖRNYEI Z. Motivational strategies in the language classroom[M]. Cambridge: Cambridge University Press, 2001.

[作者简介]

金伟（1994− ），女，北京人，中国科学院信息工程研究所博士生，主要研究方向为大数据访问控制与密钥管理。

李凤华（1966− ），男，湖北浠水人，博士，中国科学院信息工程研究所研究员、博士生导师，主要研究方向为网络与系统安全、信息保护、隐私计算。

余铭洁（1998− ），男，江西景德镇人，中国科学技术大学博士生，主要研究方向为大数据访问控制。

郭云川（1977− ），男，四川营山人，博士，中国科学院信息工程研究所正研级高级工程师、博士生导师，主要研究方向为访问控制、形式化方法。

周紫妍（1998− ），女，河北秦皇岛人，中国科学院信息工程研究所博士生，主要研究方向为访问控制。

房梁（1989− ），男，山西太原人，博士，中国科学院信息工程研究所副研究员，主要研究方向为信息安全、访问控制。