



面向行为可信的大数据安全系统形式化描述^{*}

闫 智¹, 詹 静^{1,2,3}

(1. 北京工业大学计算机学院 北京 100124; 2. 可信计算北京市重点实验室 北京 100124;
3. 信息安全等级保护关键技术国家工程实验室 北京 100124)

摘 要:大数据的存储和分析通常是在云计算环境下利用 Hadoop 计算框架进行处理,而基于云的大数据安全风险来源于云计算服务商以及远程数据应用者对未授权文件及内容的越权操作。针对云中大数据应用的安全问题,提出了面向行为可信的大数据服务系统。采用 Z 形式化方法对该系统进行描述,给出了读取、更新、添加及删除操作的抽象行为定义。根据模型中的抽象行为,划分可信行为与恶意行为,并给出了各个操作的安全分析。最后通过 Z/EVES 工具对该系统描述的正确性进行了证明和验证。

关键词:大数据;云计算安全;形式化;Z 语言;可信行为;Hadoop

doi: 10.3969/j.issn.1000-0801.2014.07.006

Formal Description of Trusted Behavior Oriented Security System for Big Data

Yan Zhi¹, Zhan Jing^{1,2,3}

(1. College of Computer Science, Beijing University of Technology, Beijing 100124, China;

2. Beijing Municipal Key Laboratory of Trusted Computing, Beijing 100124, China;

3. National Engineering Laboratory for Critical Technologies of Information Security Classified Protection, Beijing 100124, China)

Abstract: Usually the process of storing and analyzing with big data is under the framework of Hadoop within the environment of cloud computing while the risks of big data application on cloud derive from cloud service providers and remote users who access data without authorization. Based on the security issues of big data applying on cloud, a security system for big data oriented by trusted behavior was given. The system with Z formal specification was described and the definition of abstract behaviors which contains reading, updating, adding and deleting operation was given. According to the abstract behavior model, behaviors were divided into trusted behavior and malicious behavior and security analysis on each operation was given. By using Z/EVES tool verified and proved the correctness about the description.

Key words: big data, cloud computing security, formalization, Z, trusted behavior, Hadoop

1 引言

随着各个领域数据生成速度的加快,需要处理的数据量急剧膨胀。有效地分析和利用这些庞大的数据资源能够

带来巨大的价值。利用传统存储和分析工具对内容庞大、种类繁多的大数据进行处理已经不能满足其要求。Hadoop 提供了一个可靠的大数据共享存储和分析系统^[1]。它是由开源组织 Apache 在 Google 云计算的 map/reduce 和 GFS

^{*} 国家科技重大专项基金资助项目(No.2012ZX03002003),高等学校博士学科点专项科研基金资助项目(No.20131103120001)

(Google file system, Google 文件系统)思想启发下开发出的计算框架,并成功应用到诸如亚马逊、Yahoo、Facebook 等公司的应用中。而 CSP(cloud service provider, 云计算服务提供商)拥有大数据存储能力和计算资源,因此往往通过云计算平台对大数据进行处理。目前许多组织选择在租赁硬件所搭建的云端运行 Hadoop 或提供 Hadoop 服务。例如 Cloudera 提供在公共(或私有)云端运行 Hadoop 的工具; Amazon 提供 Hadoop 云服务,名为 Elastic MapReduce。

由于云计算的服务模式是数据外包,数据的所有权转移给了 CSP,云中存储的大数据通常是明文的,一旦 CSP 拥有数据的使用权,恶意的 CSP 可能会窃取用户数据,云平台亦可能受到黑客攻击致使安全机制失效或被非法控制,从而导致非授权人读取数据,给大数据安全带来了威胁。因此要保障在云端应用 Hadoop 处理大数据的安全,应从行为角度控制对数据的非法使用。

2 研究背景及相关工作

Hadoop 是当前最为人知的大数据技术实施方案。它在设计之初并未考虑过安全问题,在 Hadoop 1.0.0 或者 CDH3 版本之后,Hadoop 在安全方面加入了基于 Kerberos 的身份认证机制和基于 ACL 的访问控制机制。然而 Kerberos 的认证机制只是针对机器级别的安全认证,并未对 Hadoop 平台本身进行认证^[2]。而基于 ACL 的访问控制虽然限制了用户与组成员对 Hadoop 中资源的访问或节点间的通信,但该机制依赖于管理员对 hadoop-policy.xml 中访问列表的配置,这种基于传统的访问控制列表容易在服务器端被篡改而不易察觉。况且对于企业,访问控制列表需要经常被更新,操作过于繁琐且不易维护。因此 Hadoop 自身的安全机制是不完善的。

国内相关研究^[2]通过密码技术改善 Hadoop 的安全问题。使用基于公钥体制的身份认证,并采用了数字信封技术对传输数据进行加密,同时对存储在云端的数据也进行加密存储。但该方案在数据运算过程中仍然需要解密,在增加系统运算复杂性的同时也会产生安全问题。

Indrajit Roy 等人提出了基于强制访问控制和差分隐私技术的 Airavat 系统^[3]。该系统解决了在某种场景下 MapReduce 的安全问题,在该场景中计算程序提供者希望借助数据拥有者的数据做相关研究,这里假设程序提供者写的 mapper 函数不安全,可能会对敏感字段进行输出。Airavat 系统通过限制网络向外发送中间数据,并将标签附

在输入、输出和中间数据上用于访问控制,还预定义 mapper 函数的输出范围来保障数据安全。因此 Airavat 系统保障了对敏感数据执行不受信任的代码进行数据集的一般特性或综合特性计算的情况下的安全。这里主要保障了 MapReduce 中 map 步骤的安全。然而该系统并没有对存储在 HDFS 中的所有数据进行保护,还需要在更底层对信息流进行进一步的保护。

Sedic 系统^[4]通过对数据添加标签,使用字符串扫描器搜索预定义的敏感数据集,依据数据安全等级自动分离 MapReduce 的计算任务,在私有云中进行敏感数据的计算,在公有云中进行其他部分的计算,以此保护 MapReduce 中的数据安全。该系统有效防范了公有云的控制者对隐私数据的获取。由于加入大数拆分步骤,增加了数据处理的复杂度。

Samir Saklikar 在 2012 年 RSA 信息安全大会上提出了 MapReduce 需要内嵌安全和可信基^[5]。对 MapReduce 提出了内省架构实施可扩展的动态访问控制。他认为需要对数据进行细粒度的访问控制,在访问控制决策执行点构建基于内容的访问控制。在策略执行点部署基于内容的访问控制请求、MapReduce 任务分析、访问控制策略。该方案通过安全内省机制有效控制了 MapReduce 中不满足安全策略的行为请求的执行。该方案当前只是提出一种设想,并未真正实现。

本文以 Hadoop 平台为研究背景,从行为角度描述云计算中大数据行为模型,定义可信行为与恶意行为,通过控制恶意行为的执行,保障大数据服务安全。

3 行为模型定义

对于云计算中的行为操作,定义为两类:一类行为是合法的,例如数据拥有者对数据的处理,CSP 对用户数据的正常维护等,这类行为称为可信行为;另一类行为是非法的,例如没有操作权限的用户对文件的读取,CSP 对文件的不彻底删除等,定义为恶意行为。为了给出行为模型,需要使用通用的形式化方法描述行为,用软件形式规格说明语言——Z 语言^[6,7]对行为模型进行定义。这种基于模型的形式方法也被称为基于状态的形式方法,其基本的思想是利用已知特性的数学抽象来为目标系统的状态特征和行为特征构造模型。Z 形式化方法的优点在于它基于一阶谓词逻辑和集合论,具有严格的数学基础和描述特性,相对于自然语言具有精确无歧义性。利用 Z 语言书写规格说明,可以提高工程质量,更有利于团队设计和实现代码^[8],同时



为编写和验证计算机程序的正确性提供了依据。

3.1 状态模式

云计算中存在两类实体,主体为用户,客体为资源,用户 Person 与资源 Resource 存在关系映射。用户 Person 涉及 4 类人,owner 对应数据拥有者、user 对应数据使用者、computer 对应数据计算者、CSP 对应云服务提供商。操作的资源按照粒度粗细可以划分为文件系统 filesystem 和文件 file。

一个用户可以对多个资源进行操作,一个资源同样可以被多个用户使用。这里的关系是一个多对多的映射。具体的资源内容可以通过抽象资源与具体内容的映射来表示,这种映射是一对一的,因为每个资源只能指向其相应的资源内容,这种关系可以用部分入射函数表示。综上所述可以定义 Cloud 中的集合类型为 [Person, Resource, Content]。在 Hadoop 中,为了保障服务可靠性,对原始数据在不同节点进行了备份,资源具有相同的属主及内容,将 Resource 定义为包类型(Z 形式语言中一种特有的类型,是一种无序、可重复的数据集合类型),允许资源具有重复内容。引入模式对 Cloud 进行进一步描述。Cloud 状态模式定义如下:

Cloud
$\text{behavior} : \text{Person} \leftrightarrow \text{Resource}$ $\text{match} : \text{Resource} \rightarrow \text{Content}$ $\text{copies} : \text{bagResource}$

当对云计算进行初始化时,不存在用户以及资源,所以 behavior、match 和 copies 关系均为空集。

InitCloud
Cloud'
$\text{behavior}' = \emptyset$ $\text{match}' = \emptyset$ $\text{copies}' = \emptyset$

对于改变了 behavior 和 match 关系的任意操作的模式为:

ΔCloud
Cloud
Cloud'

如果云计算中的某些操作行为不会引起云计算状态的任何改变,把该模式定义为:

$\exists\text{Cloud}$
ΔCloud
$\theta\text{Cloud} = \theta\text{Cloud}'$

3.2 操作模式

状态模式定义完成之后,定义操作模式。Hadoop 中对资源的操作包括读、写和修改。HDFS 设计之初并不支持对

文件内容的追加,从 HDFS 2.x 开始支持对文件追加。因此这里的修改操作指的是对内容的追加以及权限的修改。操作的资源分为两类:数据资源以及权限资源,这里数据资源又分为两级(文件系统级、文件级)。将行为抽象合并为 4 类,读取(Read)、更新(Update)、添加(Add)、删除(Delete)。Read 对应所有需要显示的操作,包括文件系统大小、数量的显示,文件目录的显示,文件内容的显示等;Update 对应修改操作,包括数据资源及权限资源的修改;Add 操作包括文件的备份及文件的上传等,对应的是文件级资源;Delete 也是对文件级资源的操作。将 Hadoop 中所有操作指令^[9]对应到抽象行为中,具体见表 1。而类似 mv(移动资源到其他位置)的这种操作可以看作 Delete 与 Add 操作的组合。

表 1 抽象云计算操作

操作	功能描述	Hadoop 中对应操作指令
Read	对所有资源的读取以及显示	cat、count、du、dus、help、ls、lsr、stat、tail、text
Update	对数据资源的更改	getmerge、setrep
	对权限资源的更改	chgrp、chmod、chown
Add	文件级资源的新增操作	copyFromLocal、copyToLocal、cp、get、mkdir、moveFromLocal、moveToLocal、put、touchz
Delete	文件级资源的删除操作	expunge、rm、rmr

下面分别给出每一个操作的完整规格说明。

3.2.1 Read: 读取操作

在定义的系统可信行为是可以执行的,这里 DoRead 是可信行为模型:

DoRead
$\exists\text{Cloud}$ $p? : \text{Person}$ $r? : \text{Resource}$ $c! : \text{Content}$
$p? : \text{dom behavior}$ $r? : \text{behavior } p?$ $c! : \text{match } r?$

对于可信行为,给出成功提示信息,对于恶意行为,需要中止并给出出错提示信息。因此需要扩展提示信息并引入操作结果判定,包括允许执行及拒绝执行。定义如下:

Report :: =Success|Illegal_Person|Not_Authorized|Reach_Max_Copies|File_Not_Clear

Action :: =allow | deny

成功操作模式定义为:

SuccessResult a!: Action rep!: Report
a!: allow rep!: Success

给出 Read 操作的前置条件,见表 2。

表 2 Read 操作前置条件

操作	部分操作	前置条件
Read	DoRead	$p? \in \text{dom behavior} \wedge r? = \text{behavior } p?$
	IllegalPerson	$p? \notin \text{dom behavior}$
	NotAuthorized	$p? \in \text{dom behavior} \wedge r? \neq \text{behavior } p?$

根据前置条件,Read 操作完整规格说明可以定义为:
 $\text{Read} \triangle \text{DoRead} \vee \text{IllegalPerson} \vee \text{NotAuthorized}$, 垂直形式的模式为:

Read $\exists \text{Cloud}$ $p?: \text{Person}$ $r?: \text{Resource}$ $c!: \text{Content}$ $a!: \text{Action}$ $\text{rep}!: \text{Report}$
$p? \in \text{dom behavior} \wedge$ $r? = \text{behavior } p? \wedge$ $r? \in \text{dom match} \wedge$ $\text{behavior}' = \text{behavior} \wedge$ $\text{match}' = \text{match} \wedge$ $\text{copies}' = \text{copies} \wedge$ $a! = \text{allow} \wedge$ $\text{rep}! = \text{Success} \wedge$ $c!: \text{match } r?)$ $\vee$ $p? \notin \text{dom behavior} \wedge$ $\text{behavior}' = \text{behavior} \wedge$ $\text{match}' = \text{match} \wedge$ $\text{copies}' = \text{copies} \wedge$ $a! = \text{deny} \wedge$ $\text{rep}! = \text{Illegal_Person}$ $\vee$ $p? \in \text{dom behavior} \wedge$ $r? = \text{behavior } p? \wedge$ $\text{behavior}' = \text{behavior} \wedge$ $\text{match}' = \text{match} \wedge$ $\text{copies}' = \text{copies} \wedge$ $a! = \text{deny} \wedge$ $\text{rep}! = \text{Not_Authorized}$

3.2.2 Update:更新操作

与 Read 操作类似,DoUpdate 为可信行为,IllegalPerson 和 NotAuthorized 操作为不可信行为,不再分别描述,直接给出 Update 操作的完整规格说明: $\text{Update} \triangle \text{DoUpdate} \vee \text{IllegalPerson} \vee \text{NotAuthorized}$, Update 操作的前置条件见表 3。

表 3 Update 操作前置条件

操作	部分操作	前置条件
Update	DoUpdate	$p? \in \text{dom behavior} (r? = \text{behavior } p?)$
	IllegalPerson	$p? \notin \text{dom behavior}$
	NotAuthorized	$p? \in \text{dom behavior} \wedge r? \neq \text{behavior } p?$

根据前置条件,Update 垂直形式的操作模式为:

Update ΔCloud $p?: \text{Person}$ $r?: \text{Resource}$ $c?: \text{Content}$ $a!: \text{Action}$ $\text{rep}!: \text{Report}$
$(p? \in \text{dom behavior} \wedge$ $r? = \text{behavior } p? \wedge$ $(\text{behavior}' = \text{behavior} \oplus (p? \mapsto r?) \vee$ $\text{match}' = \text{match} \oplus \{r? \mapsto c?\}) \wedge$ $\text{copies}' = \text{copies} \wedge$ $a! = \text{allow} \wedge$ $\text{rep}! = \text{Success})$ $\vee$ $(p? \notin \text{dom behavior} \wedge$ $\text{behavior}' = \text{behavior} \wedge$ $\text{match}' = \text{match} \wedge$ $\text{copies}' = \text{copies} \wedge$ $a! = \text{deny} \wedge$ $\text{rep}! = \text{Illegal_Person}$ $\vee$ $(p? \in \text{dom behavior} \wedge$ $r? = \text{behavior } p? \wedge$ $\text{behavior}' = \text{behavior} \wedge$ $\text{match}' = \text{match} \wedge$ $\text{copies}' = \text{copies} \wedge$ $a! = \text{deny} \wedge$ $\text{rep}! = \text{Not_Authorized})$

在 Update 中 DoUpdate 对两类资源进行修改,behavior' = behavior $\oplus \{p? \mapsto r?\}$ 指的是对权限资源的更改,而 match' = match $\oplus \{r? \mapsto c?\}$ 指的是对数据资源的更改。



3.2.3 Add; 添加操作

用相同的方法定义 Add 操作模式, Add 操作的完整规格说明为:

$\text{Add} \triangleq \text{DoAdd} \vee \text{IllegalPerson} \vee \text{MaxCopies}$

Add 操作的前置条件见表 4。

表 4 Add 操作前置条件

操作	部分操作	前置条件
Add	DoAdd	$p? \in \text{dom behavior}$
	IllegalPerson	$p? \notin \text{dom behavior}$
	MaxCopies	$p? \in \text{dom behavior} \wedge \#(\{r?\} \triangleleft \text{copies}) = \text{maxcopies}$

依据前置条件, 给出 Add 垂直形式的操作模式为:

Add
$p?: \text{Person}$ $r?: \text{Resource}$ $c?: \text{Content}$ $a!: \text{Action}$ $\text{rep}!: \text{Report}$ $\text{maxcopies}: \mathbb{N}$
$(p? \in \text{dom behavior} \wedge$ $r? \neq \text{behavior } p? \wedge$ $c? \notin \text{ran match?} \wedge$ $\text{behavior}' = \text{behavior} \cup (p? \rightarrow r?) \wedge$ $\text{match}' = \text{match} \cup (r? \rightarrow c?) \wedge$ $\text{copies}' = \text{copies} \cup (r? \rightarrow 1) \wedge$ $a! = \text{allow} \wedge$ $\text{rep}' = \text{Success})$ $\vee$ $(p? \in \text{dom behavior} \wedge$ $r? = \text{behavior } p? \wedge$ $c? = \text{match } r? \wedge$ $\#(\{r?\} \triangleleft \text{copies} < \text{maxcopies} \wedge$ $\text{behavior}' = \text{behavior} \wedge$ $\text{match}' = \text{match} \wedge$ $\text{copies}' = \text{copies} \cup \{r? \rightarrow 1\}$ $a! = \text{allow} \wedge$ $\text{rep}' = \text{Success})$ $\vee$ $(p? \notin \text{dom behavior} \wedge$ $\text{behavior}' = \text{behavior} \wedge$ $\text{match}' = \text{match} \wedge$ $\text{copies}' = \text{copies} \wedge$ $a! = \text{deny} \wedge$

$\text{rep}' = \text{Illegal_Person}$
 $\vee$
 $(p? \in \text{dom behavior} \wedge$
 $r? = \text{behavior } p? \wedge$
 $\#(\{r?\} \triangleleft \text{copies} = \text{maxcopies} \wedge$
 $\text{behavior}' = \text{behavior} \wedge$
 $\text{match}' = \text{match} \wedge$
 $\text{copies}' = \text{copies} \wedge$
 $a! = \text{deny}$
 $\text{rep}' = \text{Reach_Max_Copies})$

3.2.4 Delete; 删除操作

Delete 操作的完整规格说明为: $\text{Delete} \triangleq \text{DoDelete} \vee \text{IllegalPerson} \vee \text{NotAuthorized}$, Delete 操作的前置条件见表 5。

表 5 Delete 操作前置条件

操作	部分操作	前置条件
Delete	DoDelete	$p? \in \text{dom behavior} \wedge r? = \text{behavior } p?$
	IllegalPerson	$p? \notin \text{dom behavior}$
	NotAuthorized	$p? \in \text{dom behavior} \wedge r? \neq \text{behavior } p?$
	FileNotClear	$p? \in \text{dom behavior} \wedge r? = \text{behavior } p?$

Delete 操作垂直形式的模式为:

Delete
ΔCloud $p?: \text{Person}$ $r?: \text{Resource}$ $a!: \text{Action}$ $\text{rep}!: \text{Report}$ $\text{maxcopies}: \mathbb{N}$
$(p? \in \text{dom behavior} \wedge$ $r? = \text{behavior } p? \wedge$ $(\text{behavior}' = \text{behavior} \triangleright \{r?\}$ $\text{match}' = \{r?\} \triangleleft \text{match} \wedge$ $\text{copies}' = \{r?\} \triangleleft \text{copies} \wedge$ $a! = \text{allow} \wedge$ $\text{rep}' = \text{Success})$ $\vee$ $(p? \in \text{dom behavior} \wedge$ $r? = \text{behavior } p? \wedge$ $(\text{behavior}' = \text{behavior} \triangleright \{r?\}$ $\text{match}' = \{r?\} \triangleleft \text{match} \wedge$ $\text{copies}' = \text{copies} \triangleleft \{(r? \rightarrow \text{maxcopies})\} \wedge$ $\#(\{r?\} \triangleleft \text{copies}) \neq 0 \wedge$ $a! = \text{deny} \wedge$ $\text{rep}' = \text{File_Not_Clear})$ $\vee$

$(p? \in \text{dom behavior} \wedge$
 $r? \neq \text{behavior } p? \wedge$
 $\text{behavior}' = \text{behavior} \wedge$
 $\text{match}' = \text{match} \wedge$
 $\text{copies}' = \text{copies} \wedge$
 $a! = \text{deny} \wedge$
 $\text{rep}' = \text{Not_Authorized})$

4 行为模型证明

Z/EVES^[10,11] 是加拿大 ORA 公司开发的一个 Z 支持工具,它具有规格说明编辑、语法和类型检查、定义域检查和定理证明等功能。通过使用 Z/EVES 工具,验证了所有定义的模式的正确性。验证界面如图 1 所示,左侧 Syntax 列下显示字母 Y,表示段落已被检查且无类型和语法错误;Proof 列下显示字母 Y,表示右侧的模式已被证明。

5 安全性分析

基于云的大数据安全风险的根本在于,云计算是一种委托计算,即他人委托云平台、远程完成的计算。资源所有权被转移,因此可能存在不可信但具有底层控制权的实体对未授权文件的恶意读取。网上曾披露政府部门通过云服务提供商来获取数据,就是因为云端确实存在未授权人员对数据的非法访问。Microsoft、Google 和 Facebook 等均收到过外国情报监控法庭对用户数据的请求。云服务商或政府部门等未授权者对数据的非法访问、分析均会导致隐私泄露。而 Hadoop 的自身安全机制薄弱以及明文存储、处理数据加深了安全风险。

根据对上述模型中 4 种操作模式的安全分析,可以得到 4 种不可信行为。

(1)对于 Read 操作,DoRead 为合法行为,只有用户归属

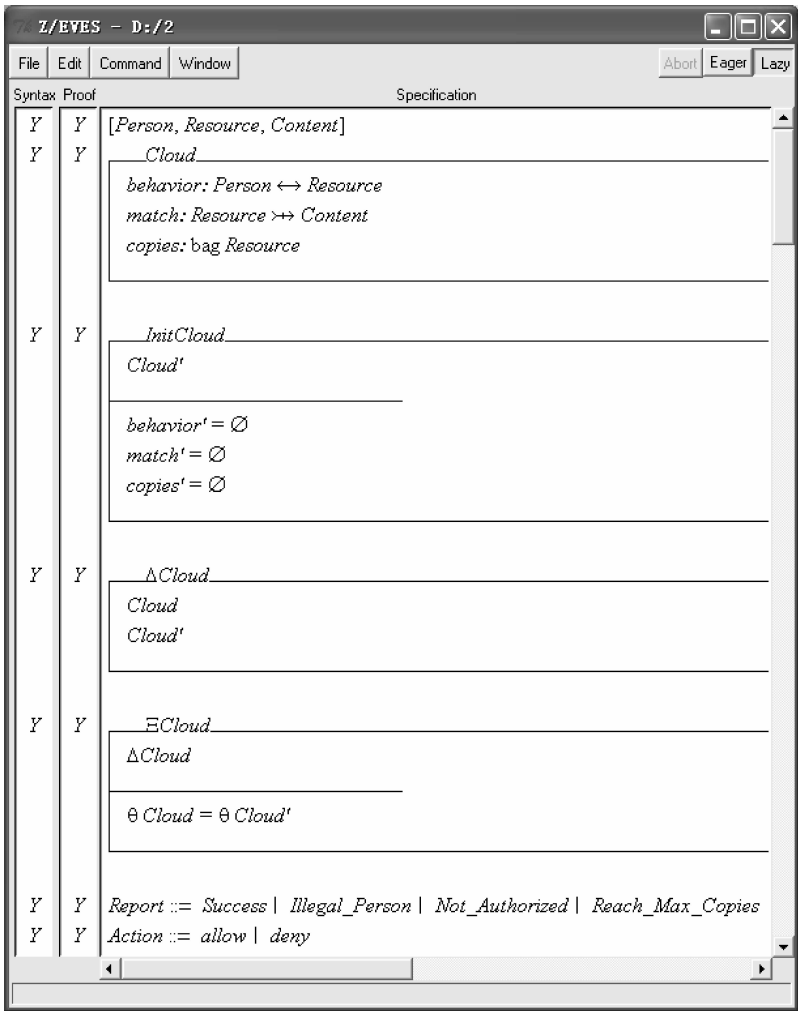


图 1 Z/EVES 工具证明界面



于 Person 类,即属于云用户,并且对请求的资源具有权限的情况下可以执行。IllegalPerson 和 NotAuthorized 为不可信行为。IllegalPerson 操作指的是云服务商创建的未授权用户、政府人员或者由于安全机制失效而引入的黑客等对资源的恶意读取。这类人员不归属 Person 类,所以请求被拒绝执行。NotAuthorized 操作针对的是归属于 Person 类的人员,其拥有对云中部分资源的操作权,但是却越权访问其他数据。

(2) 对于 Update 操作,恶意行为为 IllegalPerson 和 NotAuthorized,与 Read 操作类似,这里不再赘述。

(3) 对于 Delete 操作,DoDelete 定义为可信行为,因为模式中对 3 个关系的定义域或值域的限定删除操作要求对原始数据及备份数据进行彻底删除。然而云计算服务商可能并未真正彻底删除所有备份数据,因此存在 FileNotClear 不可信行为。在模式中体现为删除时,并未对 Resource 包中对应资源的所有副本进行删除, $\text{copies}' \neq \text{copies} \setminus \{r? \mapsto \text{maxcopies}\}$, 进行删除操作后,Resource 包中仍含有被删除资源与数量的映射 $\# (\{r?\} < \text{copies}') \neq 0$ 。IllegalPerson 和 NotAuthorized 与上述内容类似。

(4) Add 操作中,除 IllegalPerson 外,恶意行为还包括 MaxCopies 操作。因为对于存储在 Hadoop 中的文件,不允许文件副本数多于用户通过 setrep 设定的备份数量,以此防止对文件的非法复制。

对于上述安全威胁,可以针对提出的行为模型描述的恶意行为实施相应安全机制。例如,针对 IllegalPerson 和 NotAuthorized 不可信行为问题,应对大数据应用平台增加监控机制,监控机制可以利用 LSM(Linux security module, Linux 安全模块),依据行为模型设定访问控制策略,插入钩子函数,控制底层信息的访问,对可信行为予以通过,对恶意行为进行终止,遏制 CSP 及非法人员的恶意操作。针对 MaxCopies 和 FileNotClear 不可信行为问题,可以通过加密机制,对文件内容进行加密,保障内容的机密性,即使文件被非法复制或保留,也不能获取有价值的文件内容。

6 结束语

本文通过分析大数据在云计算中的应用可能引起的安全问题,提出了云端大数据行为模型,划分了可信行为与恶意行为,并提出了防止不可信行为执行,保障大数据服务安全的相关方法。该模型以数学理论为基础,通过形式化描述,精确定义出该系统的状态模式及操作模式,为后期安全策略的提出奠定了理论基础,从而指导实际系统

的安全设计。接下来将在增加模型描述粒度,完善行为模型,建立相关安全策略方面进行进一步研究。

参考文献

- 1 周品. Hadoop 云计算实战. 北京: 清华大学出版社, 2012
- 2 马媛. 基于 Hadoop 的云计算平台安全机制研究. 信息安全与保密通信, 2012(6): 89~92
- 3 Roy I, Setty S T V, Kilzer A, *et al.* Airavat: security and privacy for MapReduce. Proceedings of the 7th USENIX Symposium on Networked Systems Design and Implementation, San Jose, CA, USA, 2010
- 4 Zhang K H, Zhou X Y, Chen Y Y, *et al.* Sedic: privacy-aware data intensive computing on hybrid clouds. Proceedings of the 18th ACM Conference on Computer and Communications Security, Chicago, Illinois, USA, 2011
- 5 Saklikar S. Embedding security and trust primitives within MapReduce. http://www.emc-china.com/rsaconference/2012/en/download.php?pdf_file=TC-2003_EN.pdf, 2012
- 6 廖淮扣, 陈怡海. 软件形式规格说明语言-Z. 北京: 清华大学出版社, 2012
- 7 Spivey J M. The Z notation: a reference manual. Second Edition. UK: Prentice Hall International (UK) Ltd, 1998
- 8 Hasson P, Cooper S. A case study involving the use of Z to aid requirements specification in the software engineering course. Proceedings of the 17th Conference on Software Engineering Education and Training, Norfolk, Virginia, USA, 2004
- 9 Lam C. Hadoop in Action. 韩冀中译. 北京: 人民邮电出版社, 2011
- 10 Saaltink M. The Z/EVES system. Proceedings of the 10th International Conference of Z Users on the Z Formal Specification Notation, London, UK, 1997
- 11 Meisels I. Software manual for Windows Z/EVES Version 2.1. <http://www.docin.com/p-147668972.html>, 2000

[作者简介]



闫智,女,北京工业大学硕士生,主要研究方向为信息安全。

詹静,女,博士,北京工业大学讲师,主要研究方向为信息安全、可信计算。

(收稿日期:2014-06-22)