普适计算隐私保护策略研究

魏志强[1)] 康密军[2)] 贾东宁[1)] 殷 波[1)] 周 炜[1)]

[1)](中国海洋大学信息科学与工程学院 山东 青岛 266100)

[2)](南昌航空大学信息工程学院 南昌 330063)

摘 要 普适计算环境中,用户的隐私保护意志可以通过让用户自己制定隐私信息的访问控制策略(隐私策略)而得到实现,研究隐私策略的统一表示及其执行机制可以有效地解决隐私策略的多样性问题.文中使用多类逻辑和描述逻辑,建立了隐私策略模型和隐私策略公理,提出了隐私规则知识库的概念,给出了隐私策略的逻辑推理方法.在此之上,从应用的角度,定义了隐私策略本体,提出了隐私规则的执行流程.通过规则引擎,验证了隐私规则的有效性和可用性.

关键词 访问控制;策略;推理规则;普适计算;隐私保护

中图法分类号 TP311 **DOI 号**: 10.3724/SP.J.1016.2010.00128

Research on Privacy-Protection Policy for Pervasive Computing

WEI Zhi-Qiang[1)] KANG Mi-Jun[2)] JIA Dong-Ning[1)] YIN Bo[1)] ZHOU Wei[1)]

[1)](*College of Information Science and Engineering, Ocean University of China, Qingdao, Shandong* 266100)

[2)](*College of Information Engineering, Nanchang Hangkong University, Nanchang* 330063)

Abstract Enabling users to make privacy policy can meet their demand of protecting privacy in pervasive computing environment. In such a case, user privacy policies may be various. Therefore, it is very important to study uniform expression and execution mechanism of privacy policy. In this paper, a formal model to express user privacy policy is introduced by using many-sorted logic. Privacy policy axioms are put forward based on description logic, on the basis of these axioms, privacy rule KB is proposed that has the capacity of reasoning about privacy policy. For applying aforesaid model axioms, ontology is defined for privacy policy, and an execution mechanism is built by using inference rule technology. Finally the effect of privacy policy enforcement is verified in rule engine that supports backward chaining. Result shows the validity and availability of privacy rule in privacy-sensitive system.

Keywords access control; policy; inference rule; pervasive computing; privacy protection

1 引 言

普适计算为了向用户提供无所不在和透明访问方式的个性化服务,系统需要通过上下文感知功能,在用户无察觉或免打扰的情况下,使用用户的个人信息(即用户的上下文信息,例如身份、偏好、当前活动、当前位置、日程安排等).例如在基于位置服务(LBS)的系统中,需要实时地采集用户的位置信息,在医疗看护系统中,需要实时地采集用户的生理特

收稿日期:2009-07-15;最终修改稿收到日期:2009-09-13.本课题得到国家自然科学基金(60970130)、山东省科技攻关计划(2008GG30001010)、山东省中青年科学家奖励基金(2007BS01002)和山东省重点自然科学基金(Z2007G06)资助.**魏志强**,男,1969 年生,博士,教授,博士生导师,主要研究领域为软件工程、计算机图形图像处理和智能机器人.E-mail: weizhiqiang@ouc.edu.cn.**康密军**,男,1970 年生,博士,讲师,研究方向为智能信息系统和普适计算.**贾东宁**,男,1978 年生,硕士,讲师,主要研究方向为软件工程.**殷 波**,男,1976 年生,博士,讲师,主要研究方向为智能机器人.**周 炜**,男,1981 年生,博士研究生,主要研究方向为普适计算.

征参数(血压、心率等).在普适计算环境中,个人信息的使用过程涵盖了用户个人数据的整个生命周期,包括对其的采集、存储、传输和处理,用户隐私问题主要发生在这 4 个基本过程的采集和处理中.首先,数据采集具有覆盖范围广、采集方式不可见等特点,属于系统行为,由于采集的个人数据对用户来说具有私密性,用户对于这一过程具有隐私保护的需求;其次,以服务为目的的数据处理,其实质是个人数据被与系统交互的其它实体(软件或用户)共享的过程,对用户来说,其个人数据不可控,用户对此过程也产生隐私保护的需求,在这两种情况下,如果系统缺乏隐私保护机制,用户的个人隐私都会受到威胁,这就引发了普适计算的隐私保护问题.

隐私概念没有标准定义,但强调个人对其隐私信息的控制能力.普适计算以用户为中心,让用户制定隐私信息的访问控制策略是有效的隐私保护机制,因为系统行为仅随策略的改变而改变,有利于为用户提供灵活和自适应性的隐私信息控制接口.策略体现为授权规则,可以通过基于规则的系统实现策略的执行机制.在普适计算应用环境中,用户创建隐私策略以控制其个人数据,对于请求者的访问请求,系统根据隐私策略对其检查,如果访问被授权,请求者可以访问利用个人数据所生成的普适服务.用户一般是以自然语言描述其隐私策略,容易产生歧义性,需要研究隐私策略的统一表示问题,为了使隐私策略的效果通过应用显示出来,还需要研究隐私策略的执行机制问题.

本文第 2 节分析隐私策略原语,建立隐私策略的一阶逻辑模型;第 3 节定义隐私策略公理,并讨论隐私策略的逻辑推理方法;第 4 节给出隐私策略的本体定义,建立隐私策略的规则推理机制,并给出实验验证;第 5 节是相关研究;最后是结语,阐述本文的贡献.

2　隐私保护策略模型

以自然语言描述的隐私保护策略可以抽象为谓词逻辑公式的形式.由于普适计算应用的多样性和复杂性,反映用户隐私需求的隐私保护策略经过形式化后,所得到的谓词逻辑公式也是多样化的,没有统一的表达形式.为此,本文给出了隐私保护策略基于一阶多类逻辑的抽象模型.

2.1　隐私保护策略原语

普适计算用户以自然语言描述的隐私保护策略,虽然在表达形式和风格上具有随意性和多样性的特点,但从访问控制的语义角度来看,它们具有统一的基本结构.

定义 1.　隐私保护策略(简称为隐私策略)规定了请求者 U 在何种条件下可以获得对隐私信息 O 的访问权 A,它是一个 IF-THEN 结构,即有

$$\text{IF } condition \text{ THEN } (U,A,O).$$

定义 2.　隐私策略的基本结构由以下的 BNF 范式语法定义:

〈隐私策略〉::=〈条件〉→〈授权〉

〈条件〉::=〈约束〉{〈[且|或]〉〈约束〉}

〈授权〉::=〈操作〉{〈[且|或]〉〈操作〉}

即当隐私策略所规定的条件全部满足时,请求者可以获得对用户隐私信息的访问授权.其中,条件是各种约束的布尔表达式,授权是一组操作的布尔组合.

定义 3.　构成隐私策略的句法成份可概括为如下的策略原语(policy primitives):(1)隐私客体原语.普适计算应用系统中需要被保护的用户隐私信息称为隐私客体,记为 o_p,所有隐私客体的集合记为 O_p,关于隐私客体的一般属性用谓词 $p(o_p)$ 表示.(2)请求者原语.请求者 u_r 相对于隐私客体 o_p 的拥有者 u_o 具有某些社会关系或组织关系,用抽象的谓词 $relation(u_r,u_o)$ 表示,称为关系原语;请求者 u_r 具有的某种社会属性,用抽象的谓词 $role(u_r)$ 表示,称为角色原语.(3)拥有者原语.对隐私客体 o_p 的请求发生时,拥有者 u_o 处于某种状态,用抽象的谓词 $activity(u_o)$ 表示,称为状态原语;请求发生时拥有者 u_o 所处的地理位置,用抽象的谓词 $location(u_o)$ 表示,称为位置原语.(4)时间原语.本文使用符号 now 表示请求发生时的当前系统时间,用抽象谓词 $timeSpan(now)$ 表示请求发生时的当前系统时间在规定的时间段内.(5)授权原语.本文使用抽象的三元谓词符号 $allowed(u_r,a,o_p)$ 表示请求者 u_r 被允许在隐私客体 o_p 上执行动作 a.授权

$$(u_r,a,o_p)\in U_r\times P(A)\times O_p,$$

其中 U_r 表示请求者的集合,A 表示在隐私客体上可以执行的所有动作的集合,$P(A)$ 表示 A 的幂集.

本文采用多类逻辑(many-sorted logic)建立隐私策略的逻辑模型.与一阶逻辑不同,多类逻辑语言中的个体变项有不同的取值范围,即它们的个体域分别属于不同的类(sort).根据隐私策略基本原语,本文区分的类包括隐私客体类(*PrivacyObject*)、请求者类(*Requester*)、拥有者类(*Owner*)、访问权类(*Action*)和时间类(*Time*).对于隐私策略的描述,

上述 5 个类是很自然的选择. 由于本文使用多类逻辑来建立隐私策略的模型,在具体的应用中,如果需要加入新的类,并不改变隐私策略模型的特征和性质.

2.2 基于上下文的策略编译

在普适计算应用中,上下文(context)是一个内涵很广的概念,从隐私保护的需求来看,可以把它看作是一种包含个人知识的信息仓库,其中包括两类知识:静态上下文和动态上下文.

用户的静态上下文是对用户基本情况的描述,一般不会随着时间而变化. 在隐私策略的制定、推理和执行过程中,静态上下文信息起着关键作用,例如可以用来为策略原语赋值. 与隐私策略及其推理密切相关的用户静态上下文包括:(1)用户个人信息(profile). 例如用户姓名、电子邮件等信息,这些信息主要用来标识用户的身份,在隐私策略原语中,用户个人信息用来标识隐私客体拥有者的身份;(2)用户的社会和组织关系. 在用户知识仓库中,应该包含描述用户所拥有的社会和组织关系的信息,这对于隐私策略的制定和自动推理,是非常重要的,也是上下文信息的逻辑组织形式. (3)用户的时间段设置. 在用户的上下文中,可以包含一些逻辑上的周期性时间段,即按照用户的日常工作和生活,规定隐私客体可以被访问的时间段. 例如,可以区分工作日和休息日,工作日又可以划分时间段.

用户的动态上下文反映了用户随着时间的变化而表现为不同的状态或位于不同的地理位置等. 通过上下文知感技术,系统可以实时地获得用户的动态上下文,这是普适计算应用的特点. 将这些信息存储到用户的知识仓库中,可以用来实现动态的隐私保护机制. 例如,当用户参加私人活动时,一般不希望其社会关系或组织关系用户组中的成员查询其实时位置信息,为了制定这样的即时隐私策略,系统就要充分应用用户的动态上下文信息. 从普适计算隐私策略的角度看,用户的动态上下文主要包括两类:一个是用户的当前状态,另一个是用户的当前活动. 这两类上下文信息,可以作为隐私策略执行的约束条件.

用户制定的隐私策略,仅表达最直白的隐私意愿和约束条件,对系统来说,要想使其可执行并产生效果,要借助与应用相关的上下文信息,分解、丰富和重构用户隐私策略,使之成为可在系统中实施的隐私策略. 这一过程类似于程序设计语言的编译连接过程(如图 1 所示),将用户隐私策略看作源程序,将应用上下文看作动态连接库,将可实施的隐私策略看作编译连接成功的可执行程序. 为此,本文给出如下定义.

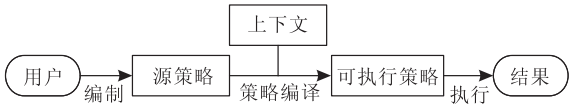

图 1 源策略、策略编译和可执行策略

定义 4. 将用户以自然语言描述、表达直接意愿的隐私策略称为源策略,根据应用的上下文信息对源策略进行命题分解和丰富,可以将其转换为可实施的隐私策略,称这一过程为策略编译,转换后的策略称为可执行策略,其基本形式为 $\bigcap P_a \rightarrow p$,其中 P_a 是原子公式,$p \in A$ 是代表授权动作的谓词符号.

2.3 可执行隐私策略模型

一个可以在系统中执行的隐私策略,必需要有相应的上下文的支持. 完备的上下文共同描述了一个隐私策略的执行环境,这些上下文信息是与具体的隐私客体、请求者、拥有者相关的. 例如,隐私策略"当我在办公室时,我的同事可以请求我的位置信息",当 Bob 请求 Alice 的位置信息时,该策略将被执行,此时的执行环境由下列的谓词公式组成.

(1) $person(alice) \wedge person(bob)$

(2) $employee(alice) \wedge employee(bob)$

(3) $colleagueOf(alice,bob) \wedge colleagueOf(alice,bob)$

(4) $working(alice) \wedge inOffice(alice)$

(5) $owner(alice) \wedge hasLocation(alice) \wedge requester(bob)$

隐私策略执行环境的完备性是否满足与具体的普适计算应用相关. 根据隐私策略原语,隐私策略执行环境模型的定义如下.

定义 5. 根据隐私策略原语定义,隐私策略执行环境 E_p 由如下的 BNF 范式定义:

$$E_p ::= \exists u_o \in U_o, \exists u_r \in U_r, \exists o_p \in O_p \left[relation(u_r,u_o) \cdots \mid role(u_r) \cdots \mid activity(u_o) \cdots \mid location(u_o) \cdots \mid p(o_p) \cdots \right],$$

其中,$p(o_p)$ 是关于隐私客体的谓词公式,与具体应用相关.

定义 6. 可执行隐私策略是由执行环境 E_p 和基于策略原语的标准隐私策略 P 组成的,即有 $E_p \wedge P$,其中

$$P ::= \forall u_o \in U_o, \forall u_r \in U_r, \forall o_p \in O_p, \forall a \in A \left[p(o_p) \wedge relation(u_r,u_o) \wedge role(u_r) \wedge activity(u_o) \wedge location(u_o) \wedge timeSpan(now) \rightarrow allowed(u_r,a,o_p) \right].$$

不失一般性,隐私策略具有如下的抽象形式:

$$\forall t_1 \in S_1, \forall t_2 \in S_2, \cdots, \forall t_m \in S_m [p_1(t_1, t_2, \cdots, t_m) \wedge p_2(t_1, t_2, \cdots, t_m) \wedge \cdots \wedge p_n(t_1, t_2, \cdots, t_m) \rightarrow allowed(u_r, a, o_p)]$$

简记为

$$\forall t_1 \in S_1, \forall t_2 \in S_2, \cdots, \forall t_m \in S_m [f \rightarrow allowed(u_r, a, o_p)],$$

其中，$S_1, S_2, \cdots, S_m$ 表示一阶多类逻辑的类，$t_1, t_2, \cdots, t_m$ 是属于不同类的项，$p(t_1, t_2, \cdots, t_m)$ 是原子公式，f 表示一阶逻辑公式.

用这种方法定义的隐私策略结构，清晰地表明了普适计算隐私策略的作用，即隐私策略是一个约束条件集合，当条件满足时，请求者被允许在隐私客体上执行某种访问权. 例如，隐私策略“我的同事可以知道我的位置”可以写为如下的一阶多类逻辑公式：

$$\forall u_o \forall u_r \forall o_p (colleagueOf(u_r, u_o) \wedge place(o_p) \wedge hasLocation(u_o, o_p) \rightarrow allowed(u_r, know(o_p), o_p)),$$

其中，u_o, u_r, o_p 分别是属于拥有者类、请求者类和隐私客体类的一个变量项；$know$ 是属于访问权类的一个算符项；$colleague(u_r, u_o)$ 是属于请求者类的一个原子公式，$hasLocation(u_o, o_p)$ 是属于拥有者类的一个原子公式，$place(o_p)$ 是属于隐私客体类的一个原子公式.

3　隐私策略的逻辑表示和推理

本节以隐私策略原语的定义为参考模型，从抽象的角度，分析和建立隐私策略的统一表示和推理方法. 基于描述逻辑理论，定义能够支持隐私策略表示的知识库，它包含普适计算应用领域结构的抽象模型和领域个体的断言形式. 隐私策略知识库的定义如下.

定义 7.　基于描述逻辑的隐私策略知识库(Privacy Knowledge Base，PKB)是一个二元组 $\mathcal{K}=(\mathcal{T}, \mathcal{A})$，其中 $\mathcal{T}$ 是 PKB 的 TBox，$\mathcal{A}$ 是 PKB 的 ABox. $\mathcal{T}$ 包含了对隐私策略原语的概念描述，而 $\mathcal{A}$ 包含了对隐私策略执行环境的概念断言和角色断言.

3.1　隐私策略的逻辑表示

隐私策略知识库 PKB 的 TBox 主要由描述普适计算应用领域结构的相关术语公理组成，根据隐私策略原语的分析，给出如下术语公理.

公理 1.　与拥有者和请求者相关的公理包括：

(1) $Owner \equiv User \sqcap \exists ownerOf.PrivacyObject$，概念 $Owner$ 表示隐私客体的拥有者的集合，其个体名字用符号 u_o 表示，其个体的概念断言为 $Owner(u_o)$，普适计算应用系统的任一用户，相对自己在系统中所拥有的隐私客体(位置信息或电子病历等)来说，均是概念 $Owner$ 的一个实例；

(2) $Owner \sqsubseteq User$，如果一个用户个体是 $Owner$ 的实例，则也是 $User$ 的一个实例；

(3) $Requester \equiv User \sqcap \exists senderOf.Query$，概念 $Requester$ 表示隐私客体的请求者的集合，其个体名字用符号 u_r 表示，其个体的概念断言为 $Requester(u_{ri})$，这里请求者是指发出查询请求的系统用户，同时也可以是自己所拥有的隐私客体的拥有者；

(4) $Requester \sqsubseteq User$，如果一个用户个体是 $Requester$ 的实例，则也是 $User$ 的一个实例.

公理 2.　具有特定属性的一组隐私客体的请求者 $AttributeBasedRequesterGroup$ 用如下的抽象概念表达式来定义：

(1) 对于实体类型的用户属性，有

$$AttributeBasedRequesterGroup \equiv Requester \sqcap \bigsqcap_{i=1}^{n} \bigsqcap_{j=1}^{m} \exists hasUserAttribute_i.UserAttributeValue_{i,j}.$$

(2) 对于字面类型的用户属性，有

$$AttributeBasedRequesterGroup \equiv Requester \sqcap \bigsqcap_{i=1}^{n} \exists hasUserLiteralAttribute_i.LiteralValue,$$

其中，$i, j \in \mathbb{N}$，$\bigsqcap \equiv \sqcap \cdots \sqcap$.

根据描述逻辑理论，基于解释 $\mathcal{I}(\Delta)$，属性用户组概念定义中表达式(1)的语义如下：

$$(AttributeBasedRequesterGroup)^{\mathcal{I}} = (Requester)^{\mathcal{I}} \sqcap \bigsqcap_{i=1}^{n} \bigsqcap_{j=1}^{m} (\exists hasUserAttribute_i.UserAttributeValue_{i,j})^{\mathcal{I}},$$

其中

$$(\exists hasUserAttribute_i.UserAttributeValue_{i,j})^{\mathcal{I}} = u_r \in \Delta^{\mathcal{I}} \mid \exists v_{ua}.(u_r, v_{ua}) \in hasUserAttribute_i^{\mathcal{I}} \wedge v_{ua} \in UserAttributeValue_{i,j}^{\mathcal{I}},$$

这里，u_r 是请求者 $Requester$ 的实例；v_{ua} 是 $UserAttributeValue_{i,j}$ 的实例，表示一个逻辑实体类型的属性值.

例 1.　属性用户组定义公理举例，在隐私策略“学校中具有教授职称的系主任可以查看我的科研履历表”中，属性用户组“具有教授职称的系主任”是一个概念，用 $DirectorWithTitleOfProessor$ 表示，则如下概念表达式

$$DirectorWithTitleOfProfessor \equiv User \sqcap \exists hasDuty.DepartmentHead \sqcap \exists hasTitle.Professor$$

表示学校中具有教授职称的系主任的这类用户的集合.其中,*hasDuty* 表示用户的职位属性,*hasTitle* 表示用户的职称属性,*DepartmentHead* 是用户属性 *hasDuty* 的属性值,*Professor* 是用户属性 *hasTitle* 的属性值.如果用户 Alice 属于这个用户组,则可以用概念断言 $DirectorWithTitleOfProessor(alice)$ 来表示 Alice 是学校中一个具有教授职称的系主任,其中 *alice* 是个体名字符号.

公理 3. 与隐私客体拥有者有特定关系的一组隐私客体的请求者记为 *RelationShipBasedRequesterGroup*,可以用如下的抽象概念表达式来定义:

(1)逻辑合取关系用户组

$$RelationShipBasedRequesterGroup \equiv Requester \sqcap \bigcap_{i=1}^{n} \exists hasRelationship_i.Owner.$$

(2)逻辑析取关系用户组

$$RelationShipBasedRequesterGroup \equiv Requester \sqcap \bigcup_{i=1}^{n} \exists hasRelationship_i.Owner.$$

(3)逻辑复合关系用户组

$$RelationShipBasedRequesterGroup \equiv Requester \sqcap \bigcup_{i=1}^{n} (\bigcap_{j=1}^{m} \exists hasRelationship_j.Owner).$$

在概念表达式中,$i,j \in \mathbb{N}$,$\bigcap \equiv \sqcap \cdots \sqcap$,$\bigcup \equiv \sqcup \cdots \sqcup$.根据描述逻辑理论,基于解释 $\mathcal{I}(\Delta)$,关系用户组概念定义(1)中表达式的语义如下:

$$(RelationShipBasedRequesterGroup)^{\mathcal{I}} = u_r \in \Delta^{\mathcal{I}} \mid \exists u_o.(u_r, u_o) \in hasRelationship_i^{\mathcal{I}} \wedge u_o \in Owner^{\mathcal{I}},$$

其中,u_r 是请求者 *Requester* 的实例;u_o 是 *Owner* 的实例.

例 2. 关系用户组定义公理举例,在隐私策略语句"在我现在的同事中,曾经是我中学同学的同事可以浏览我的工作日志"中,关系用户组"在我现在的同事中,曾经是我中学同学的同事"是一个概念 *ColleagueInSchoolmates*,可以用如下的概念表达式表示:

$$ColleagueInSchoolmates \equiv Requester \sqcap \exists colleagueOf.Owner \sqcap \exists schoolmateOf.Owner,$$

其中,*colleagueOf* 和 *schoolmateOf* 是原子角色.如果用户 Bob 属于这个用户组,则可以用概念断言 $ColleagueInSchoolmates(bob)$ 来表示 Bob 是我现在的一个同事,同时也曾经是我的中学同学,其中 *bob* 是个体名字符号.

公理 4. 概念 *TimeCondition* 表示隐私策略的抽象时间约束,其概念表达式按照拥有者状态和拥有者位置分别为

(1)在指定的时间段内当拥有者正处于某种状态时

$$TimeCondition \equiv \exists hasAllowedTimeSpan.TimeSpan \sqcap \exists hasActivity.Activity_i.$$

(2)在指定的时间段内当拥有者正处于某个位置时

$$TimeCondition \equiv \exists hasAllowedTimeSpan.TimeSpan \sqcap \exists hasLocation.Location,$$

式中,$i \in \mathbb{N}$.

公理 5. 一个已经被授权的用户组记为 *AuthorizedRequesterGroup*,用下面的概念表达式表示:

$$AuthorizedRequesterGroup \equiv Requester \sqcap \exists authorizedby.Permission.$$

基于解释 $\mathcal{I}(\Delta)$,它的语义为

$$(AuthorizedRequesterGroup)^{\mathcal{I}} = (Requester)^{\mathcal{I}} \sqcap \{u_r \in \Delta^{\mathcal{I}} \mid \exists prm.(u_r, prm) \in authorizedby^{\mathcal{I}} \wedge prm \in Permission\},$$

其中,*prm* 是访问许可 *Permission* 的实例,u_r 是请求者的个体.

公理 6. 基于用户属性的隐私策略公理为

$$\bigcap_{i=1}^{n} \bigcap_{j=1}^{m} \exists hasUserAttribute_i.UserAttributeValue_{i,j} \sqsubseteq \exists authorizedby.Permission,$$

或为

$$\bigcap_{i=1}^{n} \exists hasUserLiteralAttribute_i.LiteralValue \sqsubseteq \exists authorizedby.Permission,$$

其中,$i,j \in \mathbb{N}$.

该公理表示由用户属性表达式

$$\bigcap_{i=1}^{n} \bigcap_{j=1}^{m} \exists hasUserAttribute_i.UserAttributeValue_{i,j}$$

所确定的用户,可以被授权访问.如果请求者 u_r 满足此用户属性表达式,则称请求者 u_r 满足该隐私策略公理.因此,满足该隐私策略公理的用户,可以得到隐私客体的访问授权.这是因为,若设

$$UGrp \equiv \bigcap_{i=1}^{n} \bigcap_{j=1}^{m} \exists hasUserAttribute_i.UserAttributeValue_{i,j}$$

为上述定义的隐私策略公理中满足用户属性表达式

的所有用户的集合，同时设

$$AuGrp \equiv \exists authorizedby.Permission$$

为所有已经获得访问授权的用户的集合，则有 $UGrp \sqsubseteq AuGrp$，根据包含公理的定义可知，如果任意请求者 u_r 是 $UGrp$ 的实例，那么该请求者 u_r 也是 $AuGrp$ 的实例.

公理 7.　基于用户关系的隐私策略公理为

$$\bigcap_{i=1}^{n} \exists hasRelationship_i.Owner \sqsubseteq \exists authorizedby.Permission,$$

其中，$i \in \mathbb{N}$.

该公理表示由用户关系表达式

$$\bigcap_{i=1}^{n} \exists hasRelationship_i.Owner$$

所确定的用户，可以被授权对隐私客体的访问. 如果请求者 u_r 满足此用户属性表达式，则称请求者 u_r 满足该隐私策略公理. 因此，满足该隐私策略公理的用户，可以得到隐私客体的访问授权.

3.2　隐私策略的形式化推理

使用隐私策略公理可以表示隐私策略，ABox 中包含有用户和用户属性、关系的实例断言，就可以实现对隐私策略的推理. 隐私策略表现为授权规则，为了突出 TBox 中隐私策略公理在表示隐私策略方面的特殊性，同时为了便于模型分析，本文将隐私策略公理从 TBox 分离出来，从逻辑上区分为一个集合，将其中的隐私策略公理看作一种隐私规则，这样就形成了一个隐私规则的集合，用符号 $\mathcal{R}$ 表示. 首先给出隐私规则的定义.

定义 8.　基于描述逻辑知识库 $\mathcal{K}=(\mathcal{T},\mathcal{A})$，隐私规则是如下的包含公理：

$$C \sqsubseteq D,$$

其中，C 表示规则的前提，D 表示规则的结论，它们是 $\mathcal{T}$ 中定义的概念，对于用户个体 usr，$C \sqsubseteq D$ 表示：如果 usr 满足 C，即在 $\mathcal{A}$ 中存在一个断言 $C(usr)$，则可以推理出 usr 也满足 D，即 usr 是 D 的一个实例.

由于规则的作用可以改变隐私策略知识库的内涵，称包含有隐私规则的知识库为隐私规则知识库(privacy rule knowledge base)，它实现了对隐私策略知识库的扩展，定义如下.

定义 9.　基于描述逻辑的隐私规则知识库(Privacy Rule Knowledge Base，PRKB)是一个三元组 $\mathcal{K}=(\mathcal{T},\mathcal{A},\mathcal{R})$，其中 $\mathcal{T}$ 是 PRKB 的 TBox，$\mathcal{A}$ 是 PRKB 的 ABox，$\mathcal{R}$ 是形如 $C \sqsubseteq D$ 的隐私规则的集合，记为 RBOX. 通过基于前向推理应用规则 $\mathcal{R}$，可以产生一个扩展的 PRKB，即 $\overline{\mathcal{K}}=(\mathcal{T},\overline{\mathcal{A}})$，其中 $\overline{\mathcal{A}}$ 是 $\mathcal{A}$ 的扩展.

隐私规则的操作语义定义如下.

定义 10.　设 $\mathcal{R}$ 表示有限的规则的集合，初始的 PKB 为 $\mathcal{K}^{(0)}=\mathcal{K}=(\mathcal{T},\mathcal{A})$. $\mathcal{R}$ 中规则被应用过程中，依次产生 PKB 序列 $\mathcal{K}^{(1)},\mathcal{K}^{(2)},\cdots,\mathcal{K}^{(n)}$. 其中，$\mathcal{K}^{(i+1)}$ 是当 $\mathcal{R}$ 中存在某规则 $C \sqsubseteq D$ 使得 $\mathcal{K}^{(i)} \models C(a)$ 成立时，通过在 $\mathcal{K}^{(i)}$ 中增加一个新的断言 $D(a)$ 而得到的. 规则 $\mathcal{R}$ 应用过程终止后，最终产生的 $\mathcal{K}^{(n)}$ 与 $\mathcal{K}^{(0)}$ 相比，有相同的 TBox，但有扩展的 ABox，记为 $\overline{\mathcal{A}}$，则有 $\mathcal{K}^{(n)}=(\mathcal{T},\overline{\mathcal{A}})$.

从应用的角度看，$\mathcal{R}$ 中的隐私规则实现了基于条件的隐私授权，但从模型抽象的角度看，PRKB 中的 $\mathcal{R}$ 可以包含非隐私授权的规则，称之为通用推理规则，这类规则主要实现对 $\mathcal{A}$ 的更新，即通过规则产生新的个体断言，而这些新的个体断言可以用于隐私规则的可满足性判定.

4　隐私策略的应用实现

4.1　隐私策略的本体表示

以规则表示隐私策略形成的隐私规则，其基本组成元素与隐私策略原语是一致的，而隐私策略原语是对普适计算应用领域中的实体(概念)以及实体属性、实体间关系的精确定义. 为了使这些领域相关的概念、属性和关系不仅能够被人理解，也能够被机器理解和推理，需要建立一种共享的领域本体(Ontology)来描述这些概念、属性和关系.

隐私策略本身具有一定的结构，从形式上表现为规则，即隐私规则，本文针对隐私规则定义了相关本体，根据定义的本体，给出基于 OWL/RDF 的隐私规则的通用表示方法：

```
〈PrivacyRule〉
〈rdfs:label〉privacy rule name〈/rdfs:label〉
〈premise〉
〈logicAnd〉
〈!--查询的发送者--〉
〈condition〉
〈Query rdf:about="varQuery"〉
〈hasSender rdf:resource="varRequester"/〉
〈/Query〉
〈/condition〉
〈!-- 请求者与拥有者存在某种关系 --〉
〈condition〉
〈Query rdf:about="varOwner"〉
```

```
〈hasUserRelationship rdf:resource="varRequester"/〉
〈/Query〉
〈/condition〉
〈condition〉……〈/condition〉
〈/logicAnd〉
〈/premise〉
〈conclusion〉
〈Owner rdf:about="varOwner"〉
〈hasPrivacyObject rdf:about="varPrivacyObject"/〉
〈/Owner〉
〈/conclusion〉
〈/PrivacyRule〉
```

4.2 隐私策略的信息粒度控制

隐私策略除了可以控制请求者是否可以访问隐私客体，还可以在请求被允许的情况下，控制隐私客体的信息访问粒度，以实现隐私信息的模糊访问控制. 通过模糊隐私策略，可以极大地满足用户对隐私保护的多样化、个性化定制需求. 如图 2 所示，拥有者对同一个隐私客体 PO 制定了 3 个隐私策略，针对不同的请求者实行信息暴露的分级控制. 图中，请求者 1 满足隐私策略 1 的约束条件，因此返回的隐私信息是最精确的，而请求者 3 得到了最模糊的隐私信息描述.

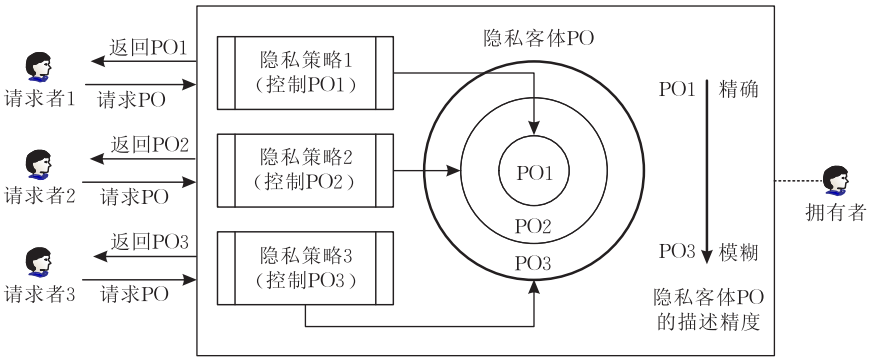

图 2 隐私策略的信息粒度控制

例如，在位置信息的隐私控制中，关于李明的位置信息 PO，可以分 3 种情况描述：(1) PO1 表示最精确的位置，即“××大学-信息学院-B 座 504”；(2) PO2 表示次精确的位置，即“××大学-信息学院”；(3) PO3 表示模糊的位置信息，即“××大学”. 一般情况下，当信息粒度具有包含关系，即 PO3⊂PO2⊂PO1，或者当隐私客体可以通过一定的二元关系(属性)表示为不同精度的信息描述时，就可以用隐私策略控制信息暴露的粒度.

通过修改隐私规则的本体表示，可以得到隐私策略粒度控制的本体表示方法. 例如，在隐私规则本体定义中再增加一个对象类型属性 granularity，其本体定义为

```
〈owl:ObjectProperty rdf:ID="granularity"〉
〈rdfs:domain rdf:resource="#PrivacyRule"/〉
〈/owl:ObjectProperty〉
```

根据这个增加的对象属性，在隐私规则本体表示中，加入一个独立的描述块，即〈granularity〉…〈/granularity〉，同时修改约束条件〈condition〉部分，就可以得到具有信息粒度控制的模糊隐私策略.

除了采用信息粒度之间的包含关系实现隐私策略的模糊化，还可以通过修改相关的本体，实现其它形式的隐私客体信息输出方法，例如在所有〈condition〉满足时，输出虚假化的隐私客体信息.

4.3 隐私规则的推理机制

在基于规则的系统中，为了实现隐私策略的执行机制，需要将基于 OWL/RDF 描述的隐私策略转换成用规则语言描述的隐私策略. 本文使用 JESS 规则语言描述隐私策略. 当隐私策略表示为 JESS 规则时，都具有图 3 所示的统一形式. 规则由两部分组成：LHS(Left-Hand Side)部分和 RHS(Right-Hand Side). 其中，规则的 LHS 部分由若干个不同的模式(pattern)组成，这些模式用来匹配规则引擎的工作区中的事实(fact)；规则的 RHS 部分执行动作(action). 对于一个规则，如果规则 LHS 中所有模式 $pattern_1$，$pattern_2$，…，$pattern_n$ 在规则引擎的工作区中都有对应的事实与之匹配时，该规则才能被激，RHS 的动作才能被执行.

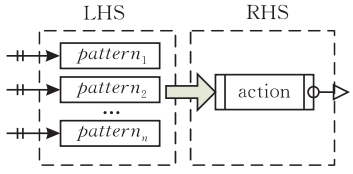

图 3 JESS 规则的抽象表示形式

根据上述内容，本文给出当系统收到请求者的查询请求时时，隐私规则在规则引擎中执行的流程（见图 4）.具体描述如下：

1. 在规则引擎中声明查询规则，由于工作区中不存在与查询规则中的模式 qfact 匹配的事实，根据后向链规则推理的原理，规则引擎会在查询事实缓冲区 QFC 中自动声明一个 need-qfact 事实；

2. need-qfact 事实与隐私规则中的第 1 个模式 need-qfact 匹配，此时隐私规则没有被激活；

3. 由于工作区中没有事实与隐私规则中的第 2 个模式 pfact 匹配，规则引擎会在隐私事实缓冲区 PFC 中自动声明一个 need-pfact 事实；

4. need-pfact 事实与映射规则中的第 1 个模式 need-pfact 匹配，但此时映射规则尚未被激活；

5. 由于在基本事实缓冲区 BFC 中已经存在 bfact 事实，并与映射规则中的第 2 个模式 bfact 匹配，此时，映射规则被激活；

6. 映射规则的 RHS 动作被执行，在 PFC 中声明 pfact 事实；

7. pfact 事实与隐私规则中的第 2 个模式 pfact 匹配，此时，隐私规则被激活；

8. 隐私规则的 RHS 动作被执行，在 QFC 中声明 qfact 事实；

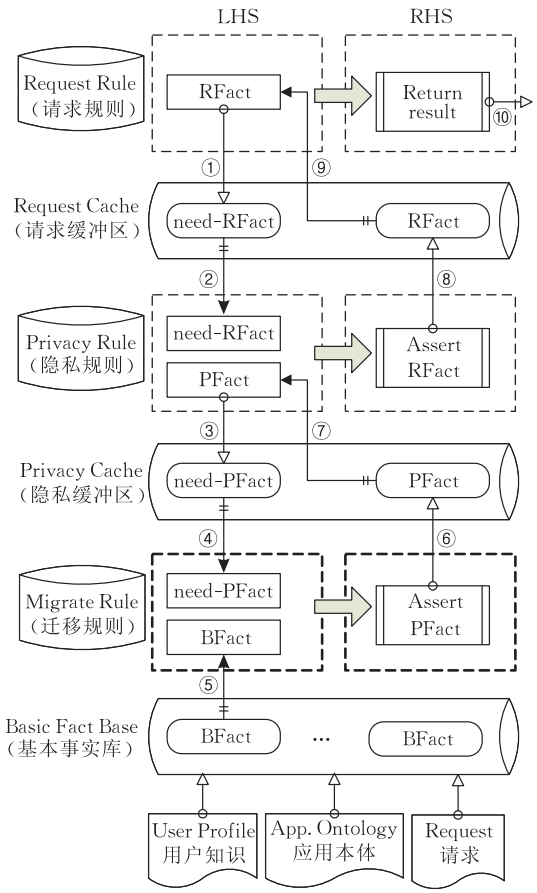

图 4　查询规则、隐私规则和映射规则的执行流程

9. qfact 事实与查询规则中的模式 qfact 匹配，查询规则被激活；

10. 查询规则的 RHS 动作被执行，输出查询结果.

4.4　隐私策略执行机制验证

实验场景：李明和张红都在软件工程研究所（SEC）工作，李明为自己的位置信息制定了隐私规则“只有同事可以访问我的位置信息”，李明的当前位置是南楼 B504，张红通过智能手机发送一个查询李明当前位置的请求.由于李明和张红都是 SEC 的成员，可以推断出他们是同事关系，因此张红的查询请求符合李明的隐私规定，系统应该返回查询结果（张明在 B504）给张红.

实验数据：实验场景的描述数据包括两类：一类是基本事实的声明，另一类是各种规则（隐私规则、查询规则、映射规则和通用规则）的定义，限于篇幅，本文仅给出基本事实的声明和隐私规则.

基本事实的声明（abox.clp）

事实模板：

```
(deftemplate bfact (slot p) (slot s) (slot o))
(deftemplate pfact (slot p) (slot s) (slot o))
(do-backward-chaining pfact)
(deftemplate qfact (slot p) (slot s) (slot o))
(do-backward-chaining qfact)
```

基本信息：

```
(assert
(bfact (p "hasLocation") (s "李明") (o "南楼 B504")))
(assert
(bfact (p "hasMember") (s "SEC") (o "李明")))
(assert
(bfact (p "hasMember") (s "SEC") (o "张红")))
```

隐私规则（privacy.clp）

```
(defrule my_colleagues_can_know_my_location_
(declare (salience 50))
(pfact (p "hasColleague") (s ?owner) (o ?colleague))
(pfact (p "hasSender") (s ?query) (o ?colleague))
(or
(need-qfact (p "hasLocation") (s ?owner) (o ?location))
(need-qfact (p "hasLocation") (s nil) (o ?location))
(need-qfact (p "hasLocation") (s ?owner) (o nil))
(need-qfact (p "hasLocation") (s nil) (o nil)))
(pfact (p "hasLocation") (s ?owner) (o ?location))
=>
(assert
(qfact (p "hasLocation") (s ?owner) (o ?location))))
```

实验环境：PC 机（1.73GHz CPU，1GB 内存），操作系统 Microsoft Windows XP，规则引擎 JESS

(Ver 6.1p4),UltraEdit 文本编辑器等.

实验目的:通过实验,演示上述隐私规则的抽象执行流程,证明此流程是可实施的;对场景数据进行测试,验证规则的隐私保护效果.

实验结果分析:本次实验中,为了便于理解实验过程,对实验数据文件做了最大程度的精简,不管是声明事实,还是定义规则,仅使用关键的概念断言事实和角色断言事实.复杂的实验可以在此基础上,通过添加各种详细的断言事实来完成.实验过程和结果如图 5 和图 6 所示.根据应用场景的描述,实验结果是正确的,隐私规则的有效性得到了验证,说明本文设计的隐私规则执行流程是可实施的.

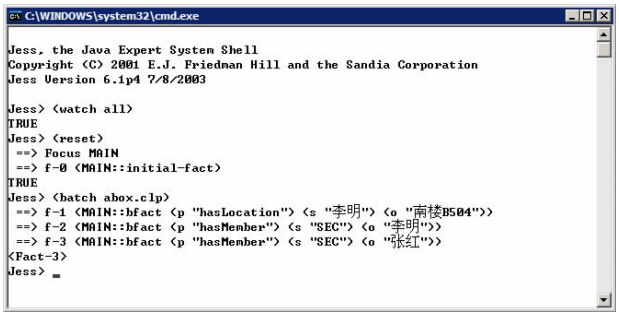

图 5 将基本信息 abox.clp 装入规则引擎的过程

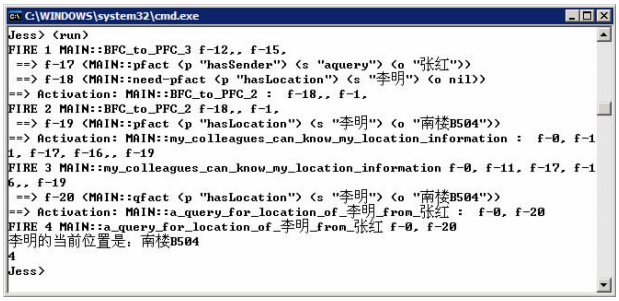

图 6 隐私规则执行结果

5 相关研究

5.1 策略的表示和推理

经典策略系统中,策略一般使用规范的策略语言定义,在语法上具有严格的形式化.但是,策略语言在语义上仍然存在一定程度的歧义性[1].从应用的角度看,很多策略语言是基于 XML 的[2].

通过采用逻辑语言的方法[3],不仅可以给出策略的形式化语法和语义,而且可以实现策略的推理[4-5].许多信任管理和访问控制领域的研究人员也采用一阶逻辑来定义安全策略语言,例如基于逻辑的安全语言 Binder[6]、授权框架语言 FAF(支持在单一系统中执行多种访问控制策略)[7]、基于角色的信任管理框架语言 RT[8].

Halpern 等采用一阶逻辑来表达和推理用于数字内容提供商的策略[1],并指出表达策略的逻辑语言必须具有足够的表达能力,以支持基于该逻辑的系统能够最大程度地理解和获取用户的策略.Halpern 的策略具有如下的通用形式:

$$\forall x_1 \cdots \forall x_m (f \Rightarrow (\neg) Permitted(t,t')),$$

其中,f 是一阶公式,t 表示策略主体,t' 表示策略动作.

在 Jajodia 提出的 ASL(Authorization Specification Language)语言[3]中,一个授权策略是一个从 4 元组(o,u,R,a)到集合$\{authorized, denied\}$的映射,其中,o,u,R,a 分别表示客体、用户、角色集合和动作.ASL 逻辑语言中定义了若干个规则,其中比较主要的是访问控制规则(Access Control Rule):

$$grant(o,u,rs,\langle sign\rangle a) \leftarrow L_1 \& \cdots \& L_n,$$

其中,o,u,rs,a 分别表示客体、主体、角色和动作;$\langle sign\rangle$表示正号(+)或负号(−);L_i代表原子公式 $in(s_1,s_2)$或原子公式 $typeOf(o,t)$,前者表示主体 s_1是主体 s_2的成员,后者表示客体 o 的类型为 t.例如,如果 $grant(o,u,R,+a)$为真,表示角色为 R 的用户 u 将被允许在客体 o 上执行动作 a.后来,Jajodia又对 ASL 语言进行了谓词扩展[7],对组用户之间的继承关系、客体间的关系和用户间的管理关系引入了分层的结构化方法.Barker[9] 采用与 Jajodia相似的方法定义如下形式的 RBAC 策略:

$$H \leftarrow L_1, L_2, \cdots, L_m,$$

其中,H 是策略的头部,$L_1, L_2, \cdots, L_m$等价于 $L_1 \wedge L_2 \wedge \cdots \wedge L_m$.

Siewe[10]针对策略提出两种授权规则:

(1)符号授权规则(signed authorization rule),其形式如下所示:

$$f \mapsto autho^+(s,o,a) \quad \text{(肯定授权规则)},$$

$$f \mapsto autho^-(s,o,a) \quad \text{(否定授权规则)},$$

其中,s 表示主体,o 表示客体,a 表示动作,f 表示任何逻辑公式,例如,

$$(in(s_1,s_2) \wedge autho^+(s_2,o,a)) \mapsto autho^+(s_1,o,a)$$

表示如果主体 s_1是组 s_2的成员,则 s_1继承 s_2的已经获得肯定授权;

(2)授权执行规则(authorization enforcement rule),其形式如下所示:

$$f \mapsto autho(s,o,a),$$

其中,s 表示主体,o 表示客体,a 表示动作,f 表示任何逻辑公式,授权执行规则主要用来实现策略的执

行机制.

5.2　隐私保护方法

从隐私数据保护的方式来看，隐私保护的研究大体上可以分为两类：自由访问型隐私保护和受限访问型隐私保护.

自由访问型隐私保护，主要针对用户数据可被任意实体自由访问的情况，由于访问者无须认证和被授权，一般采用信息隐藏的方法，例如匿名法. Langheinrich 根据公平信息原则提出了设计具有隐私保护功能的普适计算系统时应该考虑的六项指导原则[11]，其中的匿名或假名原则可以使普适计算用户不必担心因真实身份暴露而引发的隐私问题. 根据这一原则，Beresford 等构建了 MIX 网络[12]，其基础设施提供匿名服务，它在一个 MIX 区域中对服务使用者的信息进行延迟和重新排序来达到混淆观察者的目的. 普适计算的网络路由信息容易导致用户跟踪，为了获得用户身份的隐私保护，Federrath 在 MIX 的基础上，提出了一种不保护用户身份（例如电话号码）而只保护位置信息的匿名性策略[13]. 每次使用普适服务时更改用户的假名是一种保护个人身份信息的基本方法，Jendricke 设计了一个通用的身份管理框架[14]，通过该框架，用户根据不同的情况采用不同的身份，实现了用户可控的隐私保护，但是用户在选择不同的虚拟身份时会有使用负担.

受限访问型隐私保护，是指通过限制访问者对数据的访问来保护用户隐私，合法的访问者必须得到授权和认证，一般采用基于访问控制的方法. Duan 针对普适计算环境数据保护提出了数据判断方法[15]，核心思想是将访问权限嵌入到要保护的数据中，以一种自然的方式定义访问策略和机制. 虽然这种方法比较有效，但是它并不是一种用于普适计算环境隐私数据保护的完整解决方案. Heiber 为上下文感知计算提出一个隐私框架[16]，该模型从假想的隐私侵犯者角度考虑如何提供隐私保护解决方案，用户在使用普适计算系统完成其任务时，会在系统中留下各种用户数据，而隐私侵犯者根据其个人能力和对系统的访问控制权限，有可能获得对这些数据中的部分数据的访问能力，隐私侵犯者对这部分数据应用一定的推理规则，可以从中得到数据用户的隐私信息，而用户则可以根据事先确定的隐私保护条件来检查是否受到隐私侵害.

6　结　语

本文从普适计算应用环境中如何保护用户隐私信息出发，以用户隐私策略为研究对象，研究了隐私策略在模型和应用两个层面上的表示方法和推理机制；分析了构成隐私策略的各种策略原语，并将其作为用于描述隐私策略的多类逻辑的类；建立了隐私策略执行环境的模型，在此基础上，给出了可执行的隐私策略模型；基于描述逻辑，结合隐私策略原语，定义了隐私策略公理，提出了隐私规则知识库的概念，并分析了隐私策略的形式化推理过程. 我们定义了隐私规则的本体，从较为抽象和泛化的层面，提出了隐私策略基于本体的一般表示方法和基于规则的推理机制，同时给出了隐私信息的模糊控制方法，最后验证了隐私策略的执行效果.

致　谢　对审稿人提出的有益建议表示感谢！

参　考　文　献

[1] Halpern J Y, Weissman V. Using first-order logic to reason about policies//Proceedings of the IEEE Computer Security Foundations Workshop. NY, 2003: 187-201

[2] Damiani E, Di Vimercati S C, Paraboschi S, Samarati P. A fine-grained access control system for XML documents. ACM Transactions on Information and System Security, 2002, 5(2): 169-202

[3] Jajodia S, Samarati P, Subrahmanian V S. A logical language for expressing authorizations//Proceedings of the IEEE Symposium on Security and Privacy. Oakland, CA, 1997: 31-43

[4] Pucella R, Weissmann V. Reasoning about dynamic policies//Proceedings of FoSSaCS-7, 2004: 453-467

[5] Dougherty D J, Fisler K, Krishnamurthi S. Specifying and reasoning about dynamic access-control policies//Bhalla S ed. Lecture Notes in Computer Science 4130. Springer, 2006: 632-646

[6] DeTreville J. Binder, a logic-based security language//Proceedings of the IEEE Symposium on Security and Privacy. Berkeley, CA, 2002: 105-113

[7] Jajodia S, Samarati P, Sapino M L, Subrahmanian V. Flexible support for multiple access control policies. ACM Transactions on Database Systems, 2006, 26(2): 214-260

[8] Li N, Mitchell J C, Winsborough W H. Design of a role-based trust-management framework//Proceedings of the IEEE Symposium on Security and Privacy. Berkeley, CA, 2002: 114-130

[9] Barker S. Security policy specification in logic//Proceedings of the International Conference on Artificial Intelligence. Las Vegas, Nevada, USA, 2000: 143-148

[10] Siewe F, Cau A, Zedan H. A compositional framework for access control policies enforcement//Proceedings of the ACM

Workshop on Formal Methods in Security Engineering. Washington, 2003: 32-42

[11] Langheinrich M. Privacy by design — Principles of privacy-aware ubiquitous systems//Proceedings of the 3rd International Conference on Ubiquitous Computing, 2001: 273-291

[12] Beresford A R, Stajano F. Location Privacy in Pervasive Computing. IEEE Pervasive Computing, 2003, 2(1): 46-55

[13] Federrath H, Jerichow A, Pfitzmann A P. MIXes in mobile communication systems: Location management with privacy//Lecture Notes in Computer Science 1174. Springer, 1996, 1174: 121-135

[14] Jendricke U, Kreutzer M, Zugenmaier A. Pervasive privacy with identity management//Proceedings of the Workshop Security. UbiComp, 2002: 1593-1599

[15] Duan Y, Canny J. Protecting user data in ubiquitous computing environments: Towards trustworthy environments//Lecture Notes in Computer Science, Springer, 2005, 3424: 167-185

[16] Heiber T, Marron P J. Exploring the relationship between context and privacy. International Series in Engineering and Computer Science, 2005, 780: 35-48

WEI Zhi-Qiang, born in 1969, Ph. D., professor, Ph. D. supervisor. His research interests include software engineering, computer image analysis and intelligent robot.

KANG Mi-Jun, born in 1970, Ph. D., lecturer. His research interests include intelligent information system and pervasive computing.

JIA Dong-Ning, born in 1978, M. S., lecturer. His research interests focus on software engineering.

YIN Bo, born in 1976, Ph. D., lecturer. His research interests focus on intelligent robot.

ZHOU Wei, born in 1981, Ph. D. candidate. His research interests focus on pervasive computing.

Background

Pervasive computing system comprises heterogeneous computing devices that are 'invisibly' embedded into environment and provide users with ubiquitous access to services. For using these services, ubiquitous computing devices may form context-aware networks for capturing contexts about users. Such contexts can be used by pervasive computing system to adapt its functionality and behavior to various user preferences. This means pervasive computing system may facilitate unobtrusive access, manipulation, and presentation of personal data derived from contexts. The unobtrusive features of ubiquitous computing may foster unethical use of the technology but, more significantly, they are also much more conducive to inadvertent intrusions on privacy.

Privacy is the claim of individuals to determine for themselves when, how, and to what extent information about them is communicated to others. Accordingly, this paper address privacy issues by enabling individual (policy-author) to make privacy policies for controlling personal data. In such a case, entity (individual or agent) can access policy-author's personal data only if permitted by her privacy policy. This paper focuses on the representation and reasoning of user privacy policy both in the level of abstract model and application.

In this paper, privacy policy model based on many-sorted logic is introduced to uniform the privacy policy primitive which is the essential element constructing the privacy policy and provides consistent research object and privacy policy semantic. A privacy policy knowledge base PKB (TBox, ABox) including the abstract model of pervasive computing application structure and the privacy policy with the form of individual assertion is established based on description logic. The general expression of privacy policy based on ontology is presented from a relatively abstract and general level. Additionally, a proper improvement is designed to add a functionality to control the granularity of privacy information. According to the policy primitive, the expression method of privacy policy rules is presented, and layered implementation environment is defined.