基于属性的支持策略本体推理的访问控制方法研究

倪 川 黄志球 王珊珊 黄传林
(南京航空航天大学计算机科学与技术学院 南京 210016)

摘 要 基于属性的访问控制模型(ABAC)特别适用于大规模分布式网络。然而,由于网络环境的异构性以及策略控制的复杂性,其访问控制策略集往往庞大且缺乏统一语义,策略管理也因此变得复杂和易于出错。针对以上问题,使用本体一致性推理对现有的基于XACML的ABAC授权框架进行扩展:首先,对几种主要的访问控制模型在分布式环境下的性能进行量化分析;其次,通过对本体知识库的一致性检测来判断策略的一致性;最后,设计一个实验方案来验证该方法的有效性和正确性。

关键词 ABAC,语义Web,本体,XACML
中图法分类号 TP309 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.3.020

Attribute-based Access Control Method Supporting Policies Ontology Reasoning

NI Chuan HUANG Zhi-qiu WANG Shan-shan HUANG Chuan-lin
(School of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

Abstract In large-scale and distributed systems, attribute-based access control(ABAC) proves its appropriateness out of the ordinary. However, the management of policies turns out to be complex and error-prone for the heterogeneity of network environment, the complexity of policy control and policy sets of large-scale and lack-of-semantic. In order to solve the problem, this paper presented an approach based on the established XACML standard to extend current ABAC authorization architecture with ontology consistency reasoning. First, it carries out a quantitative analysis on several important access control models under distributed environment. Second, it determines the consistency of policies in accordance with the result of the consistency checking on the ontology knowledge base. Third, it designs an experimental scheme in order to verify the validity and correctness of our method.

Keywords ABAC, Semantic Web, Ontology, XACML

1 引言

近年来,随着大规模分布式网络的广泛使用,网络上充斥着大量的关键服务,如电子政务、电子商务,以及大型企业中的门户网站等,而这些服务都必须有足够的安全手段来防止非授权的访问。访问控制是企业信息安全关注的重心[1]。到目前为止,诸如基于角色的访问控制RBAC[2](Role-Based Access Control)等模型被应用于这类分布式环境中。然而,由于潜在用户众多并且大多数是事先不可知的,导致这类基于用户身份的访问控制远远不能满足大规模分布式的网络环境的需求。因此,文献[3]提出了基于属性的访问控制模型ABAC(Attribute-Based Access Control)。ABAC与XACML(eXtensible Access Control Makeup Language)标准密切相关,虽然这种基于属性的方法有更好的灵活性,但也使得策略的制定和维护更为复杂。随着网络规模的增大,由于策略规模的激增以及策略异构性和分布性等特点,策略之间的语义互操作性往往不能被很好地定义,以至于出现策略冗余和冲突等不被期望的情况。

基于以上情况,我们使用语义Web[4]技术对现有的XACML授权框架进行扩展,并加入本体策略管理点和适合的一致性推理机,通过对本体知识库的一致性检测来判断策略的一致性,从而达到统一策略语义、检测策略冗余和策略冲突的目的。由于大规模分布式网络是我们所提方法的主要应用场景,因此选择RDF[5](Resource Description Framework)、OWL[6](Ontology Web Language)和SWRL[7](Semantic Web Rule Language)等语义Web技术,来构建扩展的基于XACML[8]的ABAC授权框架。

本文第2节首先介绍了常用的访问控制模型并与ABAC进行对比,接着介绍了本体推理技术;第3节分析了支持本体一致性推理的ABAC中的实体关系,并详细介绍该框架及授权过程;第4节设计实验方案,验证了本文方法的有效性和正确性;最后介绍相关工作。

2 相关理论

2.1 访问控制

自Lampson提出访问控制矩阵的概念以来,许多访问控制模型相继出现,但只有3种取得了成功并被广泛应用,它们是自主访问控制模型[9](Discretionary Access Control,

到稿日期:2014-04-30 返修日期:2014-07-11

倪 川(1985—),男,硕士生,主要研究方向为信息安全;黄志球(1965—),男,博士,教授,博士生导师,CCF会员,主要研究方向为软件度量等;王珊珊(1963—),女,博士,副教授,硕士生导师,主要研究方向为信息安全等;黄传林(1988—),男,硕士生,主要研究方向为嵌入式安全。

DAC)、强制访问控制模型[10](Mandatory Access Control, MAC)和基于角色的访问控制 RBAC。

然而,在大规模分布式网络环境下,这些传统模型有着难以克服的缺陷。以 RBAC 为例,其基本访问控制粒度过粗,并且未提及访问的环境属性,不足以精确描述访问控制过程。在访问授权过程,RBAC 往往需要创建大量角色及角色与权限之间的映射,这种开销往往随着网络规模的扩大而呈几何性增长[11],这就是所谓的"角色爆炸"问题。ABAC 以属性作为基本访问控制粒度,能很好地解决上述问题,其基本思想是将访问控制的主体、客体(资源)和环境的相关属性作为策略决策的依据,而不是直接将权限赋予访问主体,其授权模型如图 1 所示。与之相关的 XACML 是由结构化信息标准促进组织(OASIS)批准的、用于 ABAC 安全策略表达的基于 XML 的语言。

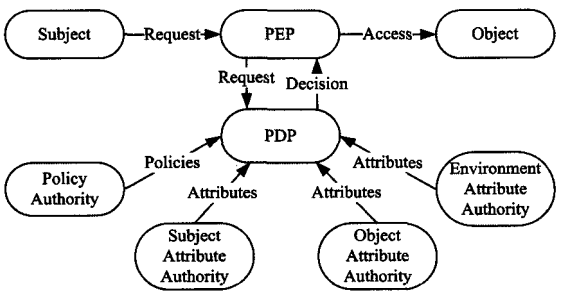

图 1 ABAC 授权框架

表 1 列出了几种主要模型在分布式环境下的性能对比。我们用{粒度(Granularity)、可扩展性(Scalability)、灵活性(Flexibility)、安全性(Security)、策略复杂性(Policy-Complexity)、策略表达力(Policy-Expression)、系统开销(System-Overhead)}7 项指标来衡量上文提到的 4 种模型。之所以选取这 7 项指标,是综合考虑了国内外访问控制方面文献的研究方向及用户需求。这些指标是分布式网络环境中访问实体对访问控制模型关注的重点,且能基本覆盖访问控制的各个方面,国内外研究也大多围绕上述一个或多个指标展开,目前为止业界还没有一个统一的量化评价标准。本文在综合各方面的基础上,尝试提出包含这 7 项指标的量化评价体系。由于访问实体对各个指标的关注程度不同,还必须设置相应的权重,根据以往研究关注的重点和用户的实际需求将权重设为{0.2,0.1,0.1,0.2,0.2,0.1,0.1},权重之和为 1。每项指标积 1 至 4 分,得分是根据这 4 种模型在分布式环境下某一指标上的表现进行排序的,表现越好,分数越高。表格最后一列 Global-Assessment 为总体评价值:

Value(Global-Assessment)=SUM(0.2 * Value(G),0.1 * Value(S),0.1 * Value(F),0.2 * Value(S),0.2 * Value(PC),0.1 * Value(PE),0.1 * Value(SO))

由表 1 可以看出,ABAC 比其他 3 种模型更适合大规模分布式网络环境。

表 1 分布式环境下主要访问控制模型的性能对比

Model	Granularity	Scalability	Flexibility	Security	Policy-Complexity	Policy-Expression	System-Overhead	Global-Assessment
DAC	1	2	3	1	4	1	1	1.9
MAC	1	1	1	4	4	1	1	2.2
RBAC	3	3	2	2	3	3	3	2.7
ABAC	4	4	4	3	1	4	4	3.2

与 RBAC 相比,ABAC 以属性作为基本授权粒度更富有表达力,更能客观地表达访问控制过程,具有更好的策略灵活性和系统可扩展性。例如,有一条自然语言规则:"拥有职务'Professor'的用户可以在每天的 8 点到 18 点访问名为'Printer'的服务。"在 RBAC 中很难表达,而在 ABAC 中该规则可表达为:

R1:can_access (s,r,e) ←(Position(s)='Professor') ∧ (Name(r)='Printer') ∧ (Time(e)>8:00 ∧ Time(e)<18:00)

属性的通用性决定了 ABAC 不用像 RBAC 那样在网络规模扩大后创建大量新角色以及角色和权限间的映射,从而解决了"角色爆炸"的问题。但是,ABAC 对 RBAC 的角色的削减是以增加策略规模、策略复杂性和管理开销为代价的[11],这种变化带来的是策略语义不一致、策略冗余和策略冲突等不被期望的结果,本文将使用本体一致性推理来解决这一问题。

2.2 本体推理

对策略本体的构建和一致性推理可以弥补 ABAC 策略语义不一致、策略冗余和策略冲突的问题。本体作为共享概念模型明确的形式化规范说明,是语义 Web 中的重要技术。因此,对某一领域构建本体后就能实现该领域语义的统一。

2006 年,Tim Berners-Lee 等人[22]提出语义 Web 的相关路线图的最新标准,其主体思想是用带有精确语义的元数据来丰富网络上人类可读的信息。本体推理的一个基本内容就是从本体知识库获得隐含的知识,从根本上说就是把隐含在显式定义和声明中的知识通过一种处理机制提取出来。但是,盲目地获得隐含的知识对建立和使用本体帮助不大,所以要先分析清楚 Web 本体推理的应用需求,才能有效地组织推理机制。本体的推理有多方面的应用:对于本体的建立者,推理的主要作用是检测冲突、优化表达和本体融合;对于本体的使用者,推理的作用主要是获得本体中的知识和运用本体中的知识解决问题。

分析和处理本体的过程发生在语义 Web 逻辑层,数据中隐含的知识可以通过推理机显式表示。常用的推理机有 RACER、Jess 等。一些简单的推理可以通过 RDFS 和 OWL 实现,而复杂的则需要专门的推理语言,如 SWRL。

上述规则 R1 用 SWRL 可表示为:

Subject(?s) ∧ hasrole(?s,'Professor') ∧ Resource(?r) ∧ hasname(?r,'Print') ∧ Enviroment(?e) ∧ lessThanOrEqual(?e,18) ∧ greaterThanOrEqual(?e,8)→Can_access(?c) ∧ hasvalue(?c,true)

3 访问授权过程

根据 ABAC 的功能划分,我们将其分为 3 个阶段进行阐述,分别为请求分析阶段、策略决策阶段和策略执行阶段,与访问控制最密切的实体分别为:用户、策略、资源和环境。它们之间的相互关系如图 2 所示。

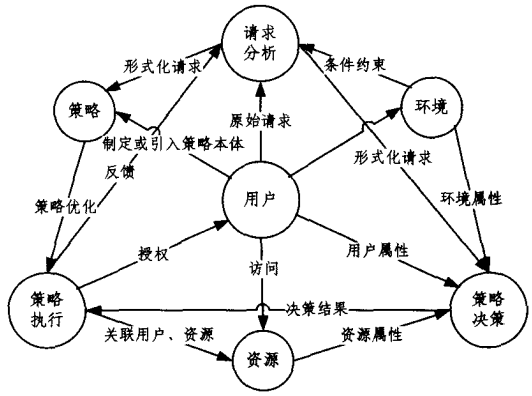

图 2　支持本体一致性推理的 ABAC 实体关系

由于 ABAC 在大规模分布式网络环境下的访问策略集过于庞大，将产生语义不一致、策略冗余和策略冲突的问题。为了解决这些不足，本文结合本体一致性推理，在 XACML 的基础上提出了支持策略本体推理的 ABAC 授权框架，如图 3 所示。用斜体和黑体表示框架的扩展部分和过程。

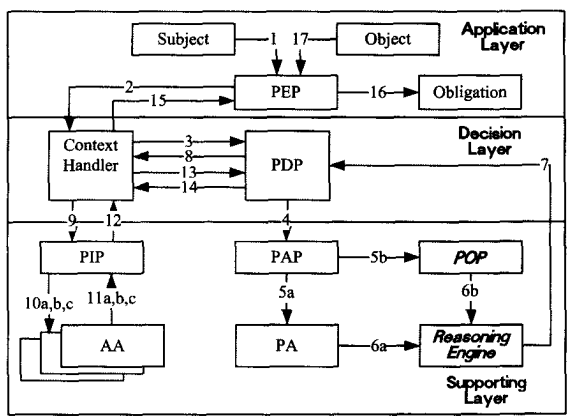

图 3　支持策略本体推理的基于属性的授权框架

文献[13]虽然尝试将 ABAC 模型像 RBAC 那样统一化处理，但是关于 ABAC 仍然没有一个标准化的授权框架。当前主流的方法是用 XACML 来描述 ABAC 授权框架。其主要功能组件阐述如下：

策略决策点 PDP(Policy Decision Point)用于对属性与访问策略的评估，并制定相应的访问控制决策。

策略执行点 PEP(Policy Enforcement Point)用于拦截访问主体发出的访问请求，并执行 PDP 做出的决策。

策略信息点 PIP(Policy Information Point)用于收集访问控制决策所需的各种属性。

策略管理点 PAP(Policy Administration Point)用于访问控制策略的制定和管理。

策略权威 PA(Policy Authority)用于存放 PAP 所制定的策略。

属性权威 AA(Attribute Authority)可分为主体属性权威、客体(资源)属性权威和环境属性权威，分别用于存放和提供访问控制所需的主体属性(如姓名、国籍)、客体(资源)属性(如资源所属的部门等)和环境属性(如允许访问的时间段等)。

上下文处理器(Context Handler)用于对整个 ABAC 授权框架工作流的管控，并对输入输出的请求和响应进行统一的 XACML 编码，使整个框架处于同一上下文背景中。

职责(Obligation)是 PEP 在收到 PDP 做出的肯定的授权决策后，在执行访问主体的访问动作之前所需满足的约束。

以上为 XACML 中所定义的 ABAC 授权框架的主要组成部分，以下为使用 Web 技术对原框架进行扩展的组件，在图 3 中用斜体表示扩展组件。

策略本体点(Policy Ontology Point，POP)为本体扩展组件，用于制定、存放和管理 OWL 策略本体。这些 ABAC 策略本体既可以由用户自定义，也可以由一些公司、国际性组织或者是使用 ABAC 技术的软件供应商来提供[14]。

推理引擎(Reasoning Engine)为本体扩展部件，用于对策略决策所需使用的策略和策略本体进行一致性推理，以策略本体知识库是否满足一致性来判断策略是否冲突。按照图 3 所示授权框架，其访问控制过程可表述如下。

3.1　请求分析阶段

这个阶段主要完成访问请求的拦截、分析、转换以及相关属性的提取等过程，这个阶段为访问决策阶段的基础，为其提供访问决策所必需的属性及确定访问决策所需的策略。

1. 用户(访问主体)向服务方发送原始访问请求，由服务方 PEP 截获请求，此请求包含了用户的主体属性。例如，用户为大学教授，要求访问网络打印机。

2. PEP 转发此请求至上下文处理器(此请求可能已包含访问所需的主体属性)。

3. 上下文处理器对该原始请求进行 XACML 编码，转化为 PDP 可识别格式，抽取其中所包含的策略决策所需属性并将其转发给 PDP。上下文处理对经过它的请求响应均进行统一编码，后续涉及代码转换的过程不再赘述。以访问主体为大学教授、访问资源为打印机、动作为打印(不考虑环境属性)为例，该请求可用 XACML 表述，如图 4 所示。

```
〈? xml version="1.0"?〉
〈Request xmlns="urn:oasis:names:tc:xacml:1.0:context"〉
  〈Subject〉
    〈Attribute AttributeId="Position" DataType="University"〉
      〈AttributeValue〉Professor〈/AttributeValue〉
        〈/Attribute〉
      〈/Subject〉
    〈Resource〉
      〈Attribute AttributeId="Equipment" DataType="University"〉
        〈AttributeValue〉Printer〈/AttributeValue〉
    〈/Attribute〉
    〈/Resource〉
      〈Action〉
        〈Attribute AttributeId="Action" DataType="University"〉
      〈AttributeValue〉print〈/AttributeValue〉
    〈/Attribute〉
  〈/Action〉
〈/Request〉
```

图 4　XACML 请求示例

4. PDP 根据该 XACML 请求中的〈Subject〉、〈Object〉和〈Action〉字段所含属性值和属性类型，来确定访问控制所需调用的策略，并向 PAP 请求相关策略。

3.2　策略决策阶段

该阶段在请求分析阶段所提供的属性的基础上，向 PIP

请求更多属性进行细粒度的访问决策，具体方法可参考文献[14]。同时，在这个阶段引入语义 Web 技术中的本体推理技术，实现策略语义统一及策略的优化。

5. PAP 创建并管理 XACML 访问控制策略，将其保存在策略权威 PA 中，PDP 根据请求向 PAP 调用相关策略集（由于 ABAC 模型策略的灵活性和策略库语义的不一致性，PAP 可能会选出不止一条符合 PDP 请求的策略），同时向 POP 调用相关策略本体。图 5 是为满足本文实验需要所构建的 OWL 策略本体（部分）。

```
〈?xml version="1.0"?〉
〈rdf:RDFxmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:swrl="http://www.w3.org/2003/11/swrl#"〉
  〈owl:Classrdf:about="urn:oasis:names:tc:xacml:1.0:policy:Subject"/〉
    〈owl:DatatypePropertyrdf:ID="urn:example:Position"〉
      〈rdfs:rangerdf:resource="http://www.w3.org/2001/XMLSchema#String"/〉
      〈rdfs:domainrdf:resource="urn:oasis:names:tc:xacml:1.0:policy:Subject"/〉
      〈rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/〉
  〈/owl:DatatypeProperty〉
  〈owl:DatatypePropertyrdf:about="urn:example:hasUserID"〉
      〈rdfs:rangerdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/〉
      〈rdfs:domainrdf:resource="urn:oasis:names:tc:xacml:1.0:policy:Subject"/〉
      〈rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/〉
  〈/owl:DatatypeProperty〉
  〈owl:DatatypePropertyrdf:about="urn:example:University"〉
      〈rdfs:rangerdf:resource="http://www.w3.org/2001/XMLSchema#String"/〉
      〈rdfs:domainrdf:resource="urn:oasis:names:tc:xacml:1.0:policy:Subject"/〉
      〈rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/〉
      〈rdfs:subClassOfrdf:resource="urn:example:hasUserID"/〉
  〈/owl:DatatypeProperty〉
...
〈/rdf:RDF〉
```

图 5　OWL 策略本体示例（部分）

6. 图 6 所示为策略一致性判断与处置流程。PA 向推理机提供 XACML 策略集，同时，POP 向推理机提供策略本体，一致性推理机此时开始推理过程。首先，是对本体知识库的推理。将策略本体映射为实事库，访问策略映射为规则库，使用通用本体知识库推理机（通用推理机更适合分布式环境）结合事实库和规则库推理得本体知识库，此时策略语义已经一致。然后，对本体知识库的一致性进行推理，用知识库是否一致来判定有无冲突和冗余。若无冲突则执行后续步骤；若有冲突则用 XACML 相应的策略合并算法进行策略合并，也可由系统管理员根据需要自定义冲突处理方法（不在本文论述范围内）。本文在利用本体对策略进行建模的基础上，提出一种基于本体一致性推理的冲突检测方法。本体在语义表示上其内部包含隐含知识，不能直接给出，需要进行推理以发现其隐含知识，从而进一步丰富原有的知识库。同时，在表达能力上，本体局限于描述逻辑，而规则能够提供较强的逻辑表达能力，因此通过使用 SWRL 描述冲突检测规则，再通过 Jess 推理机对本体和规则进行推理，就可以检测出系统中的策略冲突。以基于主体的策略为例来说明 SWRL 规则的描述方法：

在定义策略时，若一条肯定授权策略授予主体执行某个动作的权力，而另一条否定授权策略却禁止主体执行这一动作，就会发生授权策略冲突。主要特征表现为两者具有相同的主体、目标和动作。根据这一原则编写相应的规则，如下：

规则 1：SubjectBasedAuthPolicy(?x) ∧ hasSubjects(?x,?a) ∧ hasTargets(?x,?b) ∧ CanDoActions(?x,?c) ∧ SubjectBasedAuthPolicy(?y) ∧ hasSubjects(?y,?a) ∧ hasTargets(?y,?b) ∧ CannotDoActions(?y,?c)→policy-conflict(?x.?y)

同理，当一条职责策略要求主体执行某个动作，而另一条职责策略却在相反情况下要求不执行这一动作时，就会发生职责策略冗余，它们在主体、目标、动作和事件上发生了重叠。

规则 2：SubjectBasedAuthPolicy(?x) ∧ hasSubjects(?x,?a) ∧ hasTargets(?x,?b) ∧ CanDoActions(?x,?c) ∧ SubjectBasedAuthPolicy(?y) ∧ hasNoSubjects(?y,?a) ∧ hasNoTargets(?y,?b) ∧ CannotDoActions(?y,?c)→policy-redundancy(?x.?y)

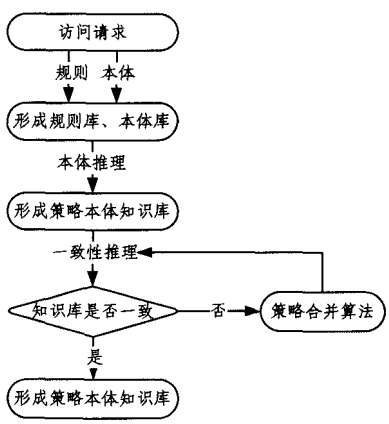

图 6　策略一致性判定与处置流程

7. 推理完成后，得到语义正确的、无冗余和冲突的策略集，并把该策略集返回给 PDP。

8. PDP 根据访问策略和用户的访问请求确定还需收集的主体、资源和环境属性集，并向上下文处理器发送属性请求。

9. 上下文处理器向 PIP 转发该属性请求。

10. PIP 向主体、资源和环境属性权威 AA 请求相关属性。

11. 主体属性权威提供主体属性（如访问打印机用户的身份、年龄等），资源属性（如被访问打印机的性能、状态等）由资源属性权威提供，环境属性（如允许访问的时间段等）由环境属性权威提供，收集完所需属性后再转发至 PIP。

12. PIP 将这些属性转发给上下文处理器。

13. 上下文处理器将这些属性转发至 PDP。

14. PDP 对所收集到的属性与相关策略进行评估，做出访问控制决策，访问控制策略可分为 permit(同意)和 deny(拒绝)两种结果，并将该决策发送给上下文处理器。

3.3 策略执行阶段

该阶段执行策略决策阶段做出的 permit/deny 策略决策。

15. 上下文处理器转发该决策给 PEP。如果该决策为拒绝访问，向访问主体发送拒绝的响应(图中未标出)。如果访问控制决策为允许，则继续以下过程。

16. PEP 必须满足执行用户访问请求的相关职责 Obligation。

17. PEP 执行用户访问控制请求，并向用户发送用 XACML 编写的安全令牌[15]。用户以该令牌为访问凭证，访问所请求资源，执行访问动作，结束后系统自动销毁安全令牌，记录访问日志，释放相关资源，等待下一次访问，控制全过程到此结束。

4 实验与分析

为了评估我们的方法，根据第 3 节提出的扩展框架设计了一个实现系统。该系统在保留本文框架大部分组件的基础上，做了适当简化，舍去了与策略冗余和冲突检测无关的组件，如 PEP 等。因此，我们认为，原始的访问请求已经是 XACML 格式的了，而不是一个未经处理的原始请求。此外，我们的系统不考虑系统应满足的职责 Obligation，原因是它与策略评估过程无关。

图 7 描述了提出的系统实现框架，其访问控制决策过程与第 3 节所述过程类似。由于分布式网络环境的异构性，选用 Jess 推理机作为本体知识库推理机，原因是其具有很好的通用性，并提供了功能全面的 API。在 Protégé 的集成环境中，通过 SWRLTab 下的 SWRLJessTab 可以将本体和相关的 SWRL 规则与 Jess 推理引擎进行集成并且支持 SWRL 插件的扩展，能够使用 SWRL 规则表达更丰富的语义[16]。

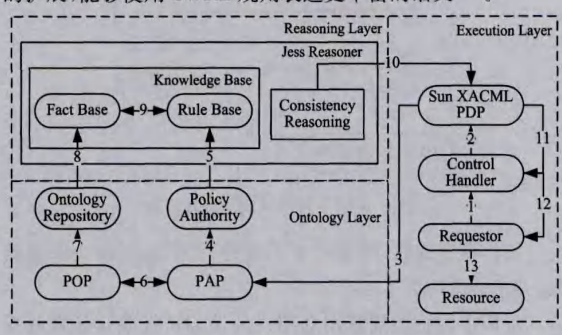

图 7 系统实现框架

我们使用带有 XACML 引用的 Sun Microsystems5(1.2 版本)作为实现系统的策略决策点 PDP，来控制系统访问控制的访问决策。

为了验证系统的正确性，选取以下 3 条 ABAC 策略，其中 R1 与 R2 冗余，R1 与 R3 冲突，R2 与 R3 冲突。

R1：can_access (s,r,e) ←(Position(s)='Professor') ∧ (Name(r)='Printer') ∧ (Time(e)>8:00 ∧ Time(e)<18:00)

R2：deny_access (s,r,e)←(Position(s)='Professor') ∧ (Name(r)='Printer') ∧ (Time(e)>18:00 ∧ Time(e)<8:00)

R3：can_access (s,r,e) ←(Position(s)='Professor') ∧ (Name(r)='Printer') ∧ (Time(e)>18:00 ∧ Time(e)<8:00)

本体 OWL 语言能很好地构建策略本体，消除策略语义不一致，但其局限于本体描述，逻辑表达能力有限。而 SWRL 语言则能够提供较强的逻辑表达能力，因此将 ABAC 规则表达为 SWRL 规则，再通过 Jess 推理机结合 SWRL 规则对本体进行一致性推理，就可以检测出系统中的策略冲突。根据 3.2 节所述，R1 与 R2 违反规则 1，R1 与 R3、R2 与 R3 违反规则 2。

对上述策略进行冲突检测，具体过程如下：

(1) 使用本体建模工具 Protégé 对规则 R1、R2、R3 建立 OWL 策略本体，如图 5 所示。

(2)使用 Protégé 内置的 SWRL Tab 定义冲突检测规则，如 3.2 节所述规则 1、规则 2(由于篇幅有限，规则不一一列举)。

(3)加载 Protégé 软件中的 Jess 推理机模块进行策略一致性推理。值得注意的是，Protégé 本身并没有一致性推理功能，必须首先在 Protégé 中加载 Jess，再将 OWL 策略本体和 SWRL 规则添加到 Jess 中，然后运行推理机，则系统将按照规则进行推理，最后将结果翻译成本体显示出来，这样就可以得到冲突检测的结果，过程如图 8 所示。

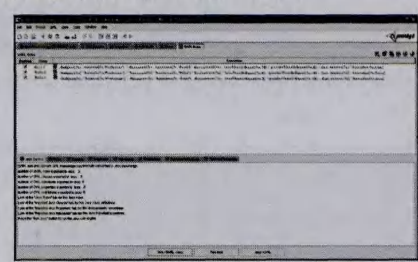

图 8 Protégé 中的 Jess 推理机调用过程

推理结论：

assert (policy-conflict rule1 rule3)

assert (policy-conflict rule2 rule3)

assert (policy-redundancy rule1 rule2)

在这个案例中，还做了有效性测试。记录了从 PDP 接受访问请求到本体策略推理完成所需的时间，并将策略的数目分别设置为 10、50、100 和 500，测试结果如表 2 所列。

表 2 不同策略复杂度所需的时间开销

策略数量(条)	时间开销(ms)	正确率%
10	145	100
50	152	98
100	157	98
500	173	99.2

测试结果显示，时间开销并未随着策略复杂度的增加而显著增加，误差率也在可接受范围之内，这表明本文方法适用于大规模分布式网络环境。系统硬件和软件环境如下：

CPU：AMD 4Cores A6-3420M APU

Memory：4GB DDR3

Operation System：Windows7 Ultimate SP1

Test Platform：GlobusToolkit4.0.5

5 相关工作

Yagüe 等人[18]首次将语义 Web 技术用于访问控制决策与执行过程。他们的方法定义了一种基于 XML 的本体策略语言，并允许动态实例化，即为获取属性值查询外部的 XML/RDF 数据源。然而，他们并没有提供类似于 OWL 语言这种更具表达力的元数据表述语言，也不包含逻辑推理能力，因此不能满足当前大规模分布式网络环境的需求。Torsten Priebe 等人[14]将语义 Web 技术与 ABAC 模型相结合，用于解决开放网络环境中异构属性方案问题，他们通过构建一个基于本体的属性推理机制，简化了 ABAC 访问控制决策所需属性的获取过程。该方法虽然使用 OWL 进行策略本体构建，且具有一定的逻辑推理能力，却没有考虑访问控制策略异构的问题，不能解决策略的语义和冲突问题，其推理能力也不足以满足复杂网络环境的需求。类似的工作有：黄凤[17]提出了一种使用 Web 本体语言 OWL 构建 RBAC 策略本体对基于角色的访问控制模型进行扩展及使用描述逻辑 DL 的 RBAC 策略表示和分析方法。但是由于 RBAC 模型不能解决复杂网络环境下的系统开销问题，该方法不适合大规模分布式网络。而我们的方法在 ABAC 模型的基础上对语义 Web 进行扩展，使得扩展后的框架能解决策略语义不一致问题。

此外，为了增强策略推理能力，还有许多文献做了这方面的研究，如 Shen 等人[19]为了加强 ABAC 的语义表达和推理能力，提出了 Web 服务下的语义感知 ABAC 模型(SABAC)。该模型使用 SWRL 语言增强了策略的推理能力，同时使用 Shibboleth 服务来保护访问决策中所需用到的敏感属性。Lorenzo Cirio 等人[20]将语义 Web 技术应用于带上下文属性的 RBAC 中，他们使用 DL 推理机对访问主体及客体进行分级并验证策略的一致性，但这种方法不能改变 RBAC 在分布式环境下的局限性，且不具备良好的策略表达力。Zha 等人[21]在 ABAC 的基础上提出 ABLC 系统来解决“未来策略冲突”问题，但其适用场景具有局限性，且缺乏语义和推理功能，也不具备知识共享性和可扩展性。

结束语 与上述工作相比，本文的贡献在于：在首次对常用的几种访问控制模型在分布式网络环境中的性能进行量化比较的情况下，使用 XACML 和本体一致性推理，设计了基于本体一致性推理技术的扩展的 ABAC 授权框架，解决了分布式环境下策略语义一致性、策略冗余和策略冲突等问题。最后设计了实现系统，并通过实验验证了本文方法的正确性和有效性。本文方法使得系统管理者可以将他们的关注重心从策略制定的合理性转移到策略决策的合理性上，直观来看，对访问控制策略的简化管理是建立在一个相对复杂的本体管理之上的，但是随着语义 Web 技术的广泛使用以及本体本身的共享性，越来越多的预先建立的本体将出现在访问控制领域(例如，电子政务和电子商务)[13]，这些本体由公司、国际性组织或者是使用 ABAC 技术的软件供应商提供。策略管理者可以在不同的场景中重用这些本体。因此，本体只需一次建立便可反复使用。通过使用本体技术，使得不同安全域之间的策略共享相同的语义背景。我们的方法是基于现有的 XACML 标准以及一些像 RDF 和 OWL 等的开放性标准，对现有的 ABAC 框架进行语义一致性扩展。XACML 的广泛使用确保了系统的可用性以及策略易用性。

下一步工作中，将进一步研究适合大规模分布式网络环境的策略冲突处理方法，设计合理的冲突处理算法；我们还将对本文所提出的量化评价体系做进一步的调研和细化；由于用户越来越关注分布式网络环境中的个人信息安全，我们将尝试并加入信任和隐私机制对本文框架进行扩展。此外，我们还将探索在军事指挥控制系统中加入该方法的可行性和有效性。

参 考 文 献

[1] Oh S, Sandhu R. A Model for Role Administration Using Organization Structure[C]// SACMAT'02. Monterey, California, USA, June 3-4, 2002

[2] Ferraiolo D F, Sandhu R, Gavrila S, et al. Proposed NIST Standard for Role-based Access Control[J]. ACM Transactions on Information and SystemsSecurity, 2001, 4(3)

[3] Priebe T, Dobmeier W, Muschall B, et al. ABAC-Ein Referenz model für attribute basierte Zugriffs kontrolle[C]// Proc. 2. Jahrestagung Fachbereich Sicherheit der Gesell schaft für Informatik (Sicherheit 2005). Regensburg, Germany, April 2005

[4] Berners-Lee T. A Roadmap to the Semantic Web[OL]. World Wide Web Consortium, September 1998. http://www.w3.org/DesignIssues/Semantic.html

[5] Resource Description Framework (RDF): Concepts and Syntax [OL]. World Wide Web Consortium, February 2004. http://www.w3.org/TR/2004/REC-rdf-concepts-2004021

[6] OWL Web Ontology Language Overview [OL]. World Wide Web Consortium, February 2004. http://www.w3.org/TR/2004/REC-owl-features-20040210

[7] SWRL: A Semantic Web Rule Language Combining OWL and RuleML[OL]. November 2003. http://www.daml.org/2003/11/swrl

[8] OASIS eXtensible Access Control Markup Language Technical Committee: eXtensible Access Control Markup Language(XACML)[OL]. http://www.oasis-open.org/committees/tc_home.php? wg_abbrev=xacml

[9] Sandhu R S. Access Control: The Neglected Frontier [C]// Pieprzyk J P, Seberry J, eds. ACISP 1996. LNCS 1172, Springer: Heidelberg, 1996: 219-227

[10] Bell D E, LaPadula L J. Secure Computer Systems: Mathematical Foundations and Model[M]. Mitre Corp., Bedford, MA, 1975

[11] Huang Jing-wei, Nicol D M, Bobba R, et al. A Framework Integrating Attribute-based Policies into Role-Based Access Control[C]// SACMAT'12. Newark, New Jersey, USA, June 2012: 20-22

[12] RDF Vocabulary Description Language 1.0: RDF Schema[OL] World Wide Web Consortium, February 2004. http://www.w3.org/TR/2004/REC-rdf-schema-20040210

[13] Jin X, Krishnan R, Sandhu R. A Unified Attribute-Based Access Control Model Covering DAC, MAC and RBAC[C]// Cuppens-Boulahia N, Cuppens F, Garcia-Alfaro J, eds. DBSec 2012. LNCS 7371, 2012: 41-55

[14] Priebe T. Supporting Attribute-based Access Control with Ontologies[C]// ARES'06. IEEE, 2006

(下转第 123 页)

有效的权限发生非预期减少或无效情况，避免管理操作不当导致权限不可用情形，从正确性和一致性角度确保授予用户的访问权限总是有效可用的；同时，资源访问权限的可用性准则确保任何持有角色 r 的用户均可使用 r 对应的权限访问指定资源，在保证访问权限的可用性的同时，还与包容性保持一致，能够为授权管理的包容性提供有效性保障。

分析可知，数据一致性、权限扩散可控和管理权限委托可管可控为简单安全性提供支撑；不存在孤立角色和权限、资源权限的可用性为简单可用性提供支撑；满足职责分离原则为限定安全性提供支撑；不存在孤立角色和权限为活性提供支撑；权限扩散可控和资源权限的可用性为包容性提供支撑。因此，基于 RBAC 的授权管理安全准则能够全面支持现有广泛采用的 Li Ninghui 教授提出的 RBAC 安全特性，进而为基于 RBAC 的授权管理的安全性提供判定标准、依据和有效保障。

结束语 RBAC 模型本身的安全是授权管理正确、有效和安全执行的关键，授权管理安全准则是验证模型安全的基础。从研究授权管理的安全性入手，分析了与基于 RBAC 的授权管理相关的安全特性，分析给出了基于 RBAC 的授权管理安全需求，从一致性、安全性和可用性 3 个方面研究了基于 RBAC 的授权管理安全准则，期望通过该安全准则，为基于 RBAC 的授权管理模型的执行过程提供一种安全性评估标准。分析结果表明，基于 RBAC 的授权管理安全准则能够全面支持现有广泛采用的 RBAC 安全特性，可为基于 RBAC 的授权管理的安全性提供判定标准、依据和有效保障。

授权管理安全准则研究对推动基于 RBAC 的授权管理模型的安全性分析和验证具有重要意义。然而，授权管理安全准则研究仅仅是模型安全判定的前提，后续研究中还需要对安全准则的正确性和有效性进行验证，并利用该准则验证现有基于 RBAC 的授权管理模型是否安全以验证其可行性。

参 考 文 献

[1] Ferraiolo D, Kuhn D R. Role-Based access control [C] // Proceedings of the 15th National Computer Security Conference. 1992: 554-563

[2] Sandhu R, Coyne E, Feinstein H, et al. Role-based Access Control Models[J]. IEEE Computer, 1996, 29(2): 38-47

[3] Ferraiolo D, Sandhu R, Guirila S, et al. Proposed NIST Standard for Role-based Access Control[J]. ACM Transactions on Information and System Security, 2001, 4(3): 224-274

[4] Munawer Q, Sandhu R S. Simulation of the augmented typed access matrix model (ATAM) using roles[C] // Proceedings of INFOSECU99 International Conference on Information and Security. 1999

[5] Crampton J. Authorizations and antichains [D]. Thesis, Birbeck College, University of London, UK, 2002

[6] Koch M, Mancini L V, Parisi-Presicce F. Decidability of safety in graph based models for access control[C] // Proceedings of the 7th European Symposiumon Research in Computer Security. 2002: 229-243

[7] Li N H, Mitchell J C, Winsborough W H. Beyond proof-of-compliance: Security analysis in trust management[J]. Journal of the ACM, 2005, 52(3): 474-514

[8] Li N, Tripunitara M. Security analysis in role based access control[J]. ACM Transactions on Information and System Security, 2006, 9(4): 391-420

[9] Sasturkar A, Yang P, Stoller S D, et al. Policy analysis for administrative role based access control[C] // Proceedings of the 19th IEEE Workshop on Computer Security Foundations. Washington: IEEE Computer Society, 2006: 124-138

[10] Habib M A, Abbas Q. Mutually exclusive permissions in RBAC [J]. Int. J. Internet Technology and Secured Transactions, 2012, 4(2/3): 207-220

[11] Ferrara A L, Madhusudan P, Parlato G. Security Analysis of Role-based Access Control through Program Verification[C] // Proceedings of 2012 IEEE 25TH Computer Security Foundations Symposium. 2012: 113-125

[12] Yang Ping, Gofman M, Yang Zi-jiang. Policy Analysis for Administrative Role Based Access Control without Separate Administration[C] // Wang L, Shafiq B, eds. IFIP International Federation for Information Processing 2013 (DBSec 2013). LNCS 7964, 2013: 49-64

[13] Liu Xiao-fan, Alechina N, Logan B. Expressing User Access Authorization Exceptions in Conventional Role-Based Access Control[C] // Deng R H, Feng T, eds. Springer-Verlag Berlin Heidelberg 2013(ISPEC 2013). LNCS 7863, 2013: 233-247

[14] 王婷. 面向授权管理的资源管理模型研究[D]. 郑州：信息工程大学，2011

[15] Harrison M A, Ruzzo W L, Ullman J D. Protection in operation systems[J]. Communications of the ACM, 1976, 19(8): 461-471

[16] 刘强，姜云飞，李黎明. RBAC 系统的权限泄漏问题及分析方法[J]. 计算机集成制造系统，2010，16(2)：431-438

[17] 徐璐. 基于安全标记的 Web 应用访问控制技术研究[D]. 郑州：信息工程大学，2009

(上接第 101 页)

[15] Kolter J, Schillinger R, Pernul G. A Privacy-Enhanced Attribute-Based Access Control System[C] // Data and Applications Security 2007. LNCS 4602, 2007: 129-143

[16] 葛强，沈国华，黄志球，等. Web 服务中支持本体推理的隐私保护研究[J]. 计算机科学与探讨，2013(6)：536-544

[17] 黄凤. 基于描述逻辑的访问控制策略冲突检测方法研究[D]. 南京：南京航空航天大学，2010

[18] Yagüe M, Mana A, Lopez L, et al. Applying the Semantic Web Layers to Access Control[C] // Proc. of the DEXA2003 Workshop on Web Semantics (Webs 2003). Prague, Czech Republic, September 2003

[19] Shen Hai-bo. A Semantic-Aware Attribute-Based Access Control Model for Web Services[C] // ICA3PP 2009. LNCS 5574, 2009: 693-703

[20] Cirio L, Cruz I F, Tamassia R. A Role and Attribute Based Access Control System Using Semantic Web Technologies[C] // OTM 2007 Ws. Part II, LNCS 4806, 2007: 1256-1266

[21] Zha D, Jing Ji-wu, Liu Peng, et al. Proactive Identification and Prevention of Unexpected Future Rule Conflicts in Attribute Based Access Control[C] // ICCSA 2010. Part IV, LNCS 6019, 2010: 468-481

[22] Berners-Lee T, Hall W, James A, et al. Weitzner: A framework for Web science[J]. Foundations and Trends in Web Science, 2006, 1(1): 1-130