使用控制支持的基于 XACML 的访问控制①

陶宇炜，符彦惟

(常州大学，常州 213164)

摘 要：针对网格环境下资源访问控制的特点，提出了一个基于使用控制模型 UCON，结合 XACML 和 SAML 的访问控制模型。用可扩展访问标记语言 XACML 描述访问控制的授权策略，结合 SAML 声明和请求/响应机制，根据用户、资源、环境的属性进行访问控制决策，可动态地评估访问请求，提供细粒度的访问控制和良好的互操作性。

关键词：使用控制；XACML；SAML；请求/响应；访问控制策略

Usage Control Enhanced Access Control Based on XACML

TAO Yu-Wei, FU Yan-Wei

(Network Center, Changzhou University, Changzhou 213164, China)

Abstract: Combining the feature of resource access control in the grid environment, this paper presents an access control model based on UCON, combined with XACML and SAML. The paper describes authorization policy about access control by XACML, combines SAML statement and request/response mechanism, executes access control decision based on user, resource and environment attributes, evaluates access request dynamically, and provides fine-grained access control and good interoperability.

Keywords: UCON; XACML; SAML; request/response; access control policy

网格技术将分布式异构资源通过高速网络互连，实现资源共享与协同工作。网格提供的是完全屏蔽了底层实现的、易于使用的、充分共享的资源操作平台。网格要达到资源共享的目的，必须解决资源的访问控制问题。近年来，人们在访问控制的研究方面取得了很大成果，主要的访问控制模型有自主访问控制模型和强制访问控制模型。在总结前人研究成果的基础上，Ferraiolo 和 Kuhn 在 1992 年提出了基于角色的访问控制模型(RBAC) 。通过对这些访问控制模型的研究可以看出，由于网格环境的分布式、异构性、不可控制等特点，网格跨越多个不同的地点和不同的自治域，每个自治域的访问控制策略和需求可能不同，使得网格资源的访问控制较为复杂。传统的网格访问控制机制忽略了实体之间的区分和交互，不能很好地应用于网格环境下。由于用户和资源可能在不同的域中，而且彼此之间没有了解，基于身份的访问控制也不能满足网格计算的要求。网格环境异构、动态和多域的特点决定了它需要易于扩展的、灵活的和细粒度的访问控制机制。传统的访问控制方法必须进行扩展才能移植到网格环境中。

为了满足网格环境的动态访问控制需求，在下一代访问控制模型——使用控制模型 UCON 的基础上，提出了基于 UCON 和 XACML(访问控制策略定义语言)的动态授权访问控制模型。采用 XACML 来描述 UCON 授权策略，实现策略的标准化，便于系统的扩展及不同应用系统之间的沟通，并具有很好的跨平台性。根据收集到的上下文信息动态地为用户授予权限，考虑到网格环境下资源的动态性，动态实现上下文相关的访问控制管理。为在不同实体之间建立安全通信机制，采用安全声明标记语言 SAML 在不同自治域之间交换认证、属性和授权信息。

1 相关技术分析

1.1 使用控制模型 UCON

使用控制(Usage Control)模型[1]是 J. Park 和 R.

① 收稿时间:2010-07-11;收到修改稿时间:2010-08-24

Sandhu 提出的一种新的访问控制模型，也称 ABC 模型[2]。此模型将多个因素集成到统一的框架中以克服传统的访问控制模型的不足，对传统的访问控制进行了扩充。UCON 模型包含三个基本元素：主体(Subjects)、客体(Objects)、权限(Right)和另外三个与授权有关的元素：授权规则(Authorization Rules)、条件(Conditions)、职责(Obligations)，同时提出了访问控制的连续性(continuity)和可变性(mutability)两个重要属性，如图 1 所示。

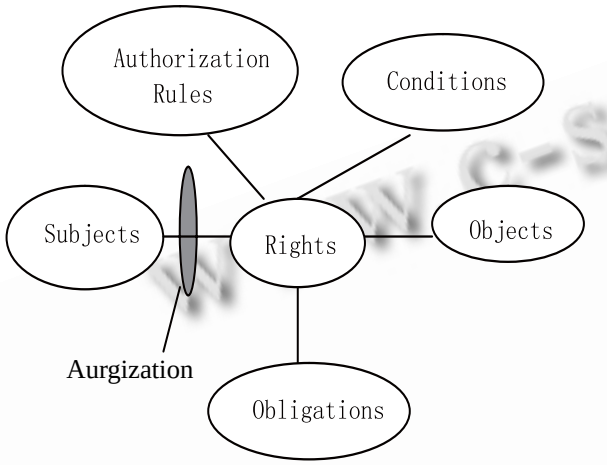

图 1 UCON 模型

主体(Subjects)、客体(Object)和权限(Right)来自传统的访问控制模型，它们的应用方式也和传统访问控制中类似。当主体提出访问请求时，授权规则(Authorization Rules)将根据主体请求的权限，检查主体和客体的安全属性，比如主体身份、角色、安全类别和客体的安全标签等，从而决定该请求是否被允许。职责(Obligation)就是在访问请求执行前或过程中必须由主体履行的行为，如用户需要访问某网站的某个资源时，该网站要求用户必须进行注册。另外，访问请求的执行必须满足某些系统和环境的约束，比如系统负载、访问时间限制等，这些就称为条件(Condition)组件，条件也称为上下文信息。在网格环境下，义务和条件也是决定访问请求是否能够被允许的至关重要因素。

在传统的访问控制模型中，授权决定仅在主体提出访问请求时起作用，而在权限执行的进程中没有任何限制。使用控制模型对控制时机进行了扩展，控制决定不但在提出访问请求时实施检查，而且在访问执行的过程中也会进行控制。

UCON 区别于传统的访问控制模型的两个显著特性是“连续性”和“可变性”。使用控制的连续性是指主体使用客体的过程中需要不间断地检查主体是否有资格进行或继续对客体的访问，即主体的访问具有一定的持续性和实时性，从而访问决策不仅出现在访问之前，它将伴随主体的整个访问过程。属性的可变性是指在主体使用客体的过程中需要更新可能发生改变的属性，属性的改变是主体使用客体的副作用造成的结果。可变性表现在主客体属性随着访问过程的不同因素(如主体的访问行为作用)而变更。可变属性的引入是 UCON 模型与其它访问控制模型的最大差别。这两个特性符合现代访问控制的需求。

1.2 XACML

XACML[3](Extensible Access Control Markup Language，可扩展访问控制标记语言)是由是国际标准化组织 OASIS 安全服务协会提出的一种通用的访问控制策略语言和访问决策语言，定义了基于 PAP(Policy Administration Point，策略管理点)、PIP(Policy Information Point，策略信息点)、PDP(Policy Decision Point，策略决策点)和 PEP(Policy Enforcement Point，策略执行点)等构件的体系结构，能在分布式的环境中根据资源、请求者和环境属性动态地评估访问请求，并进行授权决策。XACML 还定义了请求(XACML 请求)和响应(XACML 响应)格式，可构成一个问询(Query)来判断一个动作(Action)是否被允许执行，并对结果进行解释。请求和响应格式定义了 PDP 与 PEP 之间的接口。PEP 发布请求和处理响应；PDP 根据资源的访问策略、请求者的属性和环境属性等评估请求并返回一个响应。

XACML 用于表示控制信息访问的规则和策略，具备可移植、可扩展、支持参数化的策略描述和多样化的策略组合。这些规则和策略与整体访问控制和保密策略的上下文的目标资源有关。其目的是定义一个 XML 规范来表达对互联网上的信息进行访问的策略。此规范的核心思想是围绕一个四元组<subject、object、action、condition>来定义授权策略，其中 subject 代表授权实体，object 代表资源，目前对于 object 的说明还仅仅限于 XML 文档，action 代表对资源的操作，condition 是关于请求者可以对特定资源采取特定操作的先决条件的声明，用来丰富访问策略的语义。

XACML 定义的策略包括四种组件：规则(rule)、规则组合算法、职责(obligation)和目标(target)，XACML 的多数动作发生在策略中。策略比请求更具有一般性。用 XACML 来描述基于 UCON 模型的访问控制策略如下：

<Policy PolicyID=″(policy-name)″ PolicyCombinationAlg=″rule-combinging-algorithm: permit-overrides″(定义策略名称和规则组合算法)>

<Target>

<Subjects>主体及主体属性(名称、所属的域、值等)</Subjects>

<Resources>资源及资源属性(名称、所属的域、值等)</Resources>

<Actions>可执行的操作(读、写、执行、添加、修改等)</Actions>

</Target>

<Condition>上下文信息必须满足的条件(如规定 IP 地址的范围)</Condition>

<Rule effect=″permit″/>符合规则的预期后果，其值可为 permit 或 deny)

<Obligations>必须执行的职责</Obligations>

</Policy>

1.3 SAML

1.3.1 SAML 简介

SAML 是 OASIS 制定的基于 XML 实现 Web 服务安全产品之间互操作的安全访问控制框架体系和协议。SAML 建立了一种独立于协议和平台的验证和授权交换机制。具有 XML 跨平台数据表述的特性。SAML 利用 XML 对认证和授权信息进行编码，实现在 Internet 环境中异构安全系统间信息的交换和处理，从而为系统间的应用提供协同的安全服务。SAML 主要由两部分构成：声明和请求/响应协议。

声明是 SAML 的基本数据对象，是对主体(用户、角色等)的安全信息(身份、权限等)的 XML 描述形式。SAML 定义了认证、授权和属性三种声明。认证声明描述与认证成功事件相关的信息(如认证的机构、方式和有效期等)；授权决议声明描述许可权查询和检查的结果，此结果可以是接收或拒绝主体(用户、角色等)对资源的访问请求；属性声明描述与主体(用户、角色等)的认证和授权决议相关的信息(如主体的标志、所属用户组、角色、可访问的资源及权限等)。声明实际上就是一组由签发者提供的包含认证、属性和授权决议信息的集合。

请求/响应协议规定了两点间共享 SAML 数据所需交换的消息种类和格式，而两点间的消息传输通过与具体传输协议绑定实现，因此可与多种标准的传输协议或 XML 消息交换架构相绑定(如 HTTP、MIME、SMTP、FTP，以及 SOAP、Biztalk、ebXML 等)，使得 SAML 具有良好的开放性。

1.3.2 SAML 工作原理

SAML 提供了策略决策点 PDP 和策略执行点 PEP 处理用户身份验证和授权。资源的请求主体与资源提供者以 SAML 声明传递安全信息，SAML 声明由认证授权权威机构颁发，确保声明的有效性。图 2 是 SAML 的体系结构模型[4]，SAML 体系结构分析了 SAML 权威机构生成和发布声明的具体过程[5]。具体步骤为：(1)主体向身份鉴别提交凭证信息；(2)身份鉴别调用认证管理服务系统对主体凭证进行验证，并且产生认证声明，同时，通过认证的主体将得到一个含有 SAML 声明的令牌；(3)主体使用这个 SAML 令牌请求访问受保护的资源；(4)主体对被保护资源的 SAML 访问请求被策略执行点 PEP 截获，提交给策略决策点 PDP；(5)策略决策点 PDP 根据决策参考信息产生 SAML 授权声明，允许或者拒绝主体的访问请求。

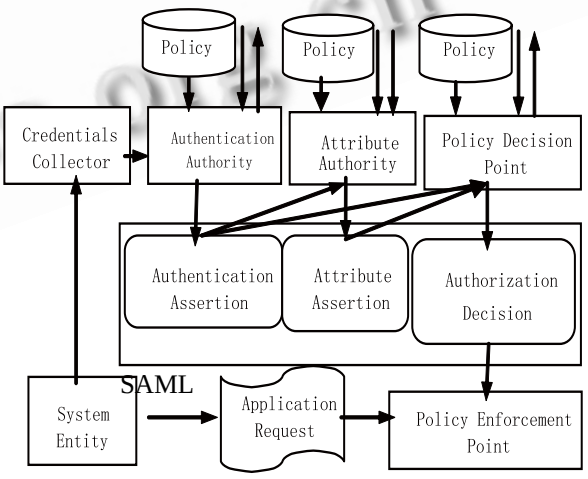

图 2 SAML 体系结构

1.4 XACML 和 SAML 的关系[6]

XACML 体系结构与 SAML 体系结构紧密相关。它们有很多相同的概念，要处理的问题域也在很大程度上重叠：验证、授权和访问控制。但是在同一问题

域中，它们要解决的是不同的问题：SAML 要解决的是验证，并提供一种机制，在合作实体间传递验证和授权决策；而 XACML 则专注于实现这些授权决策的机制。

SAML 标准提供了允许第三方实体请求验证和授权信息的接口。内部如何处理这些授权请求则由 XACML 标准解决。XACML 不但处理授权请求，而且还定义了一种机制，来创建进行授权决策所需的规则、策略和策略集的完整基础设施。

尽管 XACML 提供了一个标准化的访问控制决策模型，但它没有定义这些构件之间的通信协议和机制。需要 SAML 来定义声明、协议和传输机制。XACML 和 SAML 相结合才能提供一个完整的授权解决方案。为了更好地将 XACML 与 SAML 结合，允许 PEP 利用 SAML 请求和响应语法来完全支持 XACML 请求上下文和响应上下文。OASIS 发布了在 XACML 中使用 SAML 的文档规范[7]。规范定义了多个 SAML 扩展，并且规定了 XACML 中各构件之间的通信格式和机制。结合 SAML 的 XACML 访问控制体系结构如图 3 所示。其中 SAQ= SAMLAttributeQuery，SAS＝SAMLAttributeStatement

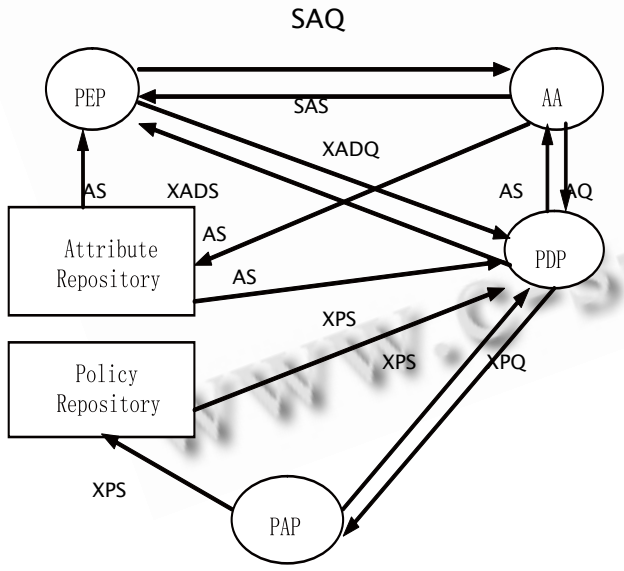

图 3 利用 SAML 的 XACML 访问控制体系结构

图 3 中有六种 Query 和 Statement，其含义和功能如下：

(1) AQ(AttributeQuery)：一种标准的 SAML 请求，用于从 AA(Attribute Authority，属性权威)请求一个或多个属性。

(2) AS(AttributeStatement)：一种标准的 SAML 声明，包含一个或多个属性。这种声明用于来自属性权威的 SAML 响应中，或者作为一种将属性存储在 Attribute Repository (属性存储器)的格式，用于 SAML 声明中。

(3) XPQ(XACMLPolicyQuery)：一种 SAML 请求的扩展，用于从策略管理点处请求一个或多个策略。

(4) XPS(XACMLPolicyStatement)：一种 SAML 声明的扩展。用于来自策略管理点的 SAML 响应中，或者作为一种将策略存储在 Policy Repository(策略存储器)的格式，用于 SAML 声明中。

(5) XADQ(XACMLAuthorizationDecisionQuery)：一种 SAML 请求的扩展，用于 PEP 从 XACML PDP 处请求授权决策。

(6) XADS(XACMLAuthorizationDecisionStatement)：一种 SAML 声明的扩展。用于来自 XACML PDP 的 SAML 响应中；也可用于用做凭证的 SAML 声明中。

有了这些 SAML 的扩展，PDP 就可以直接利用 SAML 形式的请求和响应与 XACML 的各构件之间直接通信，获取所需的信息。

2 基于使用控制UCON的网格资源动态授权访问

2.1 体系结构

访问控制体系结构如图所示，由 SAML 服务器、一个或多个属性权威(AA)、访问控制实施组件、访问控制决策组件、XACML 策略库[8](主要用来存储访问控制策略)、UDDI 服务等组件组成，如图 4 所示。

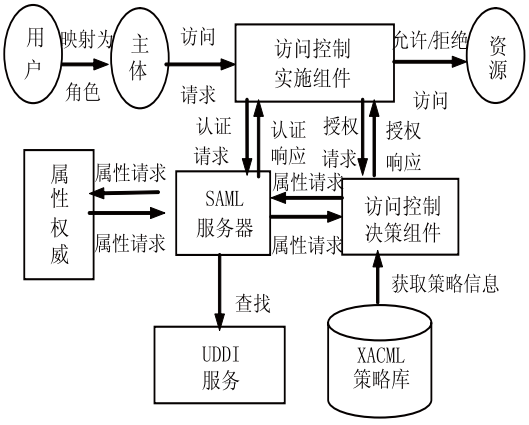

图 4 访问控制体系结构

访问控制实施组件作为受保护资源的唯一入口，用于执行使用控制策略。访问控制实施组件包括更新、定制及监视三个模块，负责监听转发请求并更新属性，实施访问控制决策组件对资源访问的最后决定，实现对资源访问的授予或拒绝。首先，访问控制实施组件从访问请求者(主体)处获取使用请求，将使用决策请求发送给访问控制决策组件，并从访问控制决策组件获取使用决策响应。其次，访问控制实施组件利用 XACML 请求语言建立一个基于主体、资源、动作和环境等属性的授权请求并向访问控制决策组件发送授权请求〈XACMLAuthorizationDecisionQuery〉、此请求中的主体身份、请求的资源(URI)和请求的操作；访问控制实施组件接收来自访问控制决策组件的授权决策(允许或拒绝对资源的访问请求)，向资源发送允许或拒绝访问的信息。必要时，还需向属性权威请求并接收需求的相关属性。

SAML 服务器负责接收来自访问控制实施组件的认证请求〈AuthnRequest〉，并产生和发送相应的认证声明。SAML 利用 UDDI 服务找到相应的属性权威；接收来自访问控制决策组件的属性请求〈AttributeQuery〉并转发给属性权威，接收来自属性权威的 SAML 属性声明，并将其转换为 XACML 的属性形式。

属性权威负责接收来自 SAML 服务器的属性请求，产生并发送相应的属性声明，其中包含若干〈AttributeStatement〉。

UDDI(Universal Description Discovery and Integration，统一描述、发现和集成协议)是一套基于 Web 的、分布式的、为 Web 服务提供信息注册中心的实现标准规范；同时也包含一组使企业能将自身提供的 Web 服务注册以使别的企业能够发现的访问协议实现标准。这种应用程序能够在后台自动访问 UDDI 目录，寻找它们需要完成业务处理的信息。UDDI 服务给相应的构件提供请求服务，以便 SAML 利用主体身份找到相应的属性权威。

访问控制决策组件包括授权、条件及职责三个模块，主要功能是依据访问控制策略、上下文信息等对主体的请求做出授权决策。授权模块与传统授权过程类似，它利用了主体和客体的信息(属性)和使用规则检查请求是否被允许。它可能返回“yes”或者“no”，它也可能返回请求的对象被授权部分的元数据和允许的权力。这些元数据随后被访问控制实施模块的定制模块用于请求数字权力。条件模块决定授权的请求是否满足条件需求(例如当前的时间、IP 地址等)，它可能限制提供的设备(如 CPU 的 ID 号，IP 地址)，时间(例如工作时间，值班时间)等。职责模块决定是否履行一定的职责，在请求使用之前，或者在请求使用的过程中履行职责。如果存在需要履行的职责，则由监视模块监视，结果的更新由 PEP 的更新模块负责。访问控制决策组件根据主体、资源和环境的属性，以及资源的访问策略(此策略以 XACMLPolicyQuery 和 XACMLPolicyStatement 的方式从策略管理点获取)，向 SAML 服务器发送属性请求〈AttributeQuery〉，获取相关授权决策所需的属性。进行授权决策，并产生和发送相应的授权决策声明，包含若干〈XACMLAuthorizationDecisionStatement〉，将决策结果返回给 PEP。

引入角色概念作为用户与权限之间的中间层[9]，通过制定相对固定的角色(管理员、资源使用者、资源拥有者、资源发布者等)，建立用户和权限之间的关系。使得用户和权限发生变化时，不需要对系统进行改动。用户和角色是一个多对多的关系。用户和角色的对应关系由 RDBMS 数据库确立。角色同时与权限相关联，不同的角色所对应的访问权限是不同的。当用户访问资源的时候，系统首先检测用户是否具有合法的身份，再通过 RDBMS 确定用户所对应的角色，根据角色来确定用户具体能访问资源的权限，决定他能对资源进行哪些操作。同时，角色具有层次关系，处于较高层次的角色拥有下层角色的权限，而下层角色则没有上层角色所拥有的权限。如管理员不但拥有资源使用者的所有权限，而且还拥有作为管理员的特定权限，而这些权限是资源使用者所没有的。

2.2 基于 XACML 的访问控制实现

我们通过 XACML 来描述主体(角色、用户等)对客体(资源、进程等)的访问权限，实现不同角色对不同资源访问的权限控制管理。当主体提出访问资源请求的时候，通过生成 SAML_request 实现主体对资源的访问请求。一个 SAML_request 以 HTTP 方式通过 SOAP(简单对象访问协议)发送，然后通过 XACML 来确定主体对资源的访问权限，并生成 SAML_response 来确定主体具有的访问权限。SOAP 是一种允许运行在不同操作系统的程序之间使用 XML 或 HTTP 进行

信息交换的机制。

用 XACML 描述如下：

```
<Subjects>
<NameIdentifier>
<SecurityDomain>http://www.ci.com</SecurityDomain>
<Name>sq</Name>
</NameIdentifier>
<Attribute>
<AttributeName>role</AttributeName>
<AttributeNamespace>http://lib.cczu.edu.cn</AttributeNamespace>
<AttributeValue>resource_owner</AttributeValue>
</Attribute>
</Subjects>
<Objects>
<Attribute>
<AttributeName>Resource</AttributeName>
<AttributeNamespace>http://lib.cczu.edu.cn</AttributeNamespace>
<AttributeValue>http://lib.cczu.edu.cn/resource</AttributeValue>
</Attribute>
</Objects>
<AuthorizationDecisionStatement>
<Decision>Permit</Decision>
</AuthorizationDecisionStatement>
<Actions>execute</Actions>
```

在<Subjects>中描述了主体属性，包括了主体名、主体所属的域、主体的值；定义了用户名和用户所属的域，以及用户所对应的角色名和角色所属的域名。在<Objects>中描述了主体所要访问的客体的属性，包括客体名、客体所属的域和客体的值。在<AuthorizationDecisionStatement>中描述了是否允许该用户访问客体，同意的话为 Permit，拒绝的话为 Denny。<Actions>中定义了用户对资源的可执行操作。

这样，我们通过 SAML 处理访问资源的请求/响应的交换。一个 SAML 请求包括提出请求的主体和客体的信息。XACML 收到一个 SAML 请求后就根据事先制定的规则或策略来判断是否允许请求使用某项资源，然后产生一个 SAML 应答，SAML 应答描述了请求者能对资源访问的权限。

3 结束语

访问控制是网格计算环境中解决安全问题的关键技术。考虑到网格环境异构、动态和多域的特点，将网格的访向控制建立在使用控制模型 UCON 的基础上，引入角色作为用户和权限的中间层，通过 RDBMS 将用户映射为与其对应的角色，根据所对应的角色取得对网格资源的访问权限。通过 XACML 描述的访问控制策略确定用户拥有的访问权限。利用 SAML 所具有的 XML 的平台和语言独立性的优势，实现分布式体系安全服务的互操作性，为异构环境中的跨域授权管理提供了可行的解决方案。能够满足网格环境下访问控制的安全性要求。

参考文献

1 Park J, Sandhu R.Towards Usage Control Models: Beyond Traditional Access Control. Proc. of the 7th ACM Symposium on Access Control Models and Technologies. 2002.

2 Sandhu R, Park J. Usage Control: A Vision for Next Generation Access Control, MMM-ACNS 2003. http://www.list.gmu.edu/conference_papers.htm

3 OASIS.Extensible access control markup language committee draft 04.2004-12.

4 OASIS.Security Assertion Markup Language(SAML) V1.1[2003-08].http://www.oasis-open.org/specs/index.php#saml

5 Wang J, Vecchio DD, Humphrey M. Extending the Security Assertion Markup Language to Support Delegation for Web Services and Grid Services. ICWS2005. Orlando:IEEE, 2005. 67－74.

6 Verma M.Control information access with XACML. [2004-10-18].http://www-128.ibm.com/developerworks/xml/library/x-xacml/ index.html/sidefile1.html

7 Anderson A. Core and Hierarchical Role Based Access Control (RBAC) profile of XACML, version2.0 committee draft 01, 30 September 2004. http://docs.Oasis- open.org/xacml/access_control- xacml-2.0-rbac_profile1-spec-cd-01.pdf

8 李真,史清华,魏峰.PMI 系统中访问控制策略的分析与设计.计算机工程与设计,2006,27(5):780－781.

9 张纲,李晓林,游赣梅,徐志伟.基于角色的信息网格访问控制的研究.计算机研究与发展,2002,39(8):952－956.