面向 Web 服务的基于属性的访问控制研究*)

沈海波　洪　帆
(华中科技大学计算机学院　武汉 430074)

摘　要　Web服务是一种新的面向服务的计算模式，由于其异构性、多域性和高度动态性，它提出了独特的安全挑战。一个关键的安全挑战就是要设计有效的访问控制机制。但目前存在的访问控制机制大多是基于身份的，存在严重的管理规模和控制粒度问题。本文提出利用基于属性的访问控制(Attribute-Based Access Control，ABAC)机制来处理Web服务的访问控制问题。ABAC采用相关实体的属性进行授权决策，能解决管理规模问题，并提供细粒度的控制。另外，文中对ABAC进行了建模，讨论了其应用，最后还给出了一种实施框架。

关键词　Web服务，基于属性的访问控制，RBAC，SAML，XACML

Study on Attribute-Based Access Control for Web Services

SHEN Hai-Bo　HONG Fan
(School of Computer, Huazhong University of Science and Technology, Wuhan 430074)

Abstract　Web service is a new service-oriented computing paradigm which poses the unique security challenges due to its inherent heterogeneity, multidomain characteristic and highly dynamic nature. A key challenge in Web services security is the design of effective access control schemes. However, the most of current access control systems is based authorization decisions on subject identity, occrues serious administrative scalability and control granularity problems. In this paper, an attribute-based access control (ABAC) model is presented to address these issues. ABAC grants accesses to services based on the attributes possessed by related entities, and can provide administratively scalable alternative to identity-based authorization methods and provide fine-grained access control for Web services. Moreover, we develop a pattern for ABAC, discuss its application issues, and also describe the implementation architecture for the system in the end.

Keywords　Web services, Attribute-based access control, RBAC, SAML, XACML

1　引言

Web服务[1]是一种崭新的分布式计算模式，基于一系列开放的标准技术，如SOAP、UDDI和WSDL等，其松散耦合、语言中立、平台无关性、开放性使得它将成为下一代电子商务的框架。它通过利用现存的Web框架为不同的平台之间建立了互操作。在Web服务环境，访问控制需要跨越安全域的边界，能够在异构的系统之间实现。并且，由于Web服务的无处不在性，服务提供者通常事先无法知晓请求者的身份。与传统的集中式系统和客户-服务器环境相比，Web服务环境更具动态性和分布性，它带来了传统的安全模型不能处理的许多新的安全挑战。最主要的挑战之一是发展有效的访问控制机制，使之能捕获与安全相关的内容和上下文信息，如时间、位置、环境状态、请求者属性等，并将它们用于访问控制决策中。特别要提出的是，这些方法的控制粒度不够，它们主要针对Web服务本身来实施访问控制。但Web服务除了软件方法外，还附带相应的输入参数，参数是Web服务的一个重要特点。因此对Web服务的访问控制应建立在"服务层和参数层"这两个层次上，才能进行更细粒度的控制。传统的基于身份的访问控制机制，如访问控制列表(ACL：Access Control List)、基于角色的访问控制(RBAC：Role-Based Access Control)、基于用户的访问控制(UBAC：User Based Access Control)等，都很难适应Web服务环境。

基于属性的访问控制(ABAC)[2,3]为建立有效的Web服务的访问控制机制带来了希望。本文首先对ABAC进行了建模，接着与RBAC进行了比较，最后探讨了ABAC的使用问题，并给出了一种基于XACML(eXtensible Access Control Markup Language，扩展访问控制标记语言)的使用架构。

2　基于属性的访问控制模型

在ABAC机制中，授权决策是基于参与决策的相关实体的属性(attributes)做出的，而不仅仅是基于身份做出的。属性是指与实体相关的一些特性。这里的实体主要有3类：主体(subject)、客体(object)和环境(environment)。主体属性包括主体的身份、角色、年龄、邮政编码、IP地址、雇员职位、已验证的PKI证书等。客体属性包括客体的身份、位置(URL)、大小、值等。特别地，Web服务的输入参数可作为一种客体属性进行处理。环境属性是与事务处理关联的属性，它通常与身份无关，但适用于授权决策，如时间、日期、系统状态、安全级别等。利用主体、客体和环境的属性来定义授权，既简化了管理，又增加了灵活性。为了更清楚地说明ABAC的特性和授权机制，我们参考Priebe等[4]提出的建模方法，

*)本文得到湖北省自然科学基金项目(NO：2004ABA055)和湖北省教育厅重点项目(NO：D200531005)资助。沈海波　副教授，博士生，主要研究领域为访问控制和网络安全；洪　帆　教授，博导，主要研究方向为密码学和信息安全。

利用类图的方式，对 ABAC 进行建模，并分为基本的 ABAC 模型和扩展的 ABAC 模型。

2.1 基本的 ABAC 模型

图 1 显示了基本 ABAC 模型中的相关元素，其中发出访问请求的主体和实际被请求访问的客体都用一组属性或特征值来表示。类 Subject 描述了实际发出访问请求的实体，一个 subject 由一组属性（SubjectAttribute）值描述，它是 SubjectAttributeValue 的实例；类 SubjectAttribute 表示主体属性方案，如此类的一个实例是“年龄”。同样地，类 Object 描述了被请求访问的实体，一个 object 由一组属性（ObjectAttribute）值描述，由类 ObjectAttributeValue 表示。另外，模型中，将授权主体和客体表示成一组关于属性或特征值的声明（assertions）。也就是说，授权不是直接在主体与客体之间定义，而是在所谓的主体描述符（SubjectDescriptor）与客体描述符（ObjectDescriptor）之间定义的。一个 SubjectDescriptor 由关于主体的属性 Qualifier（属性条件）组成，如年龄＞30、位置在办公室，或邮政编码要以 43 开始等。SubjectDescriptor 可被看作是虚拟主体，从而易于与多个实际的主体相关联。同样地，ObjectDescriptor 由关于客体的属性条件组成。像属性一样，定义描述符的声明被分成 SubjectQualifier 和 ObjectQualifier 类。主体描述符和客体描述符有点像主体组和客体组，但不是由安全管理员显式地分组，而是根据它们的属性或特征值隐式地分组。

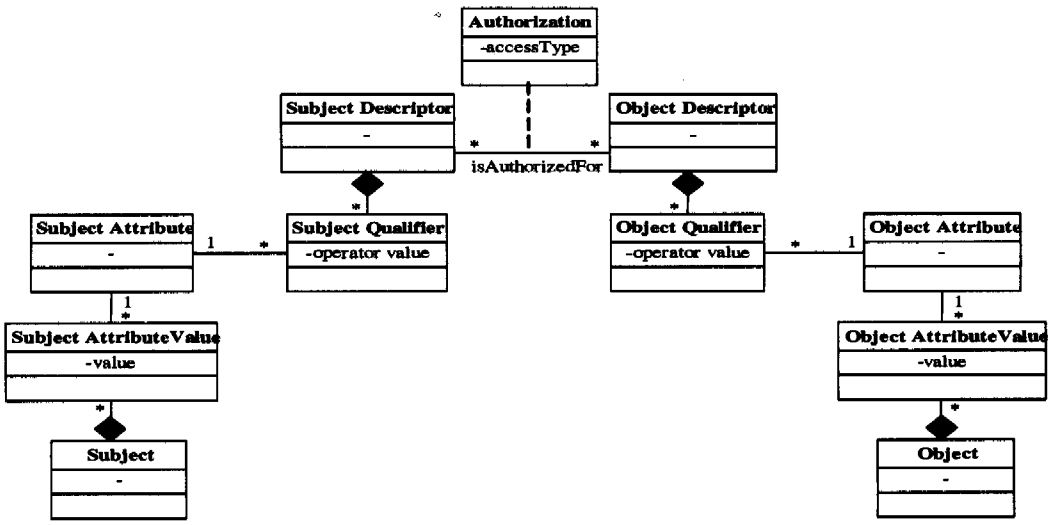

图 1 基本 ABAC 模型图

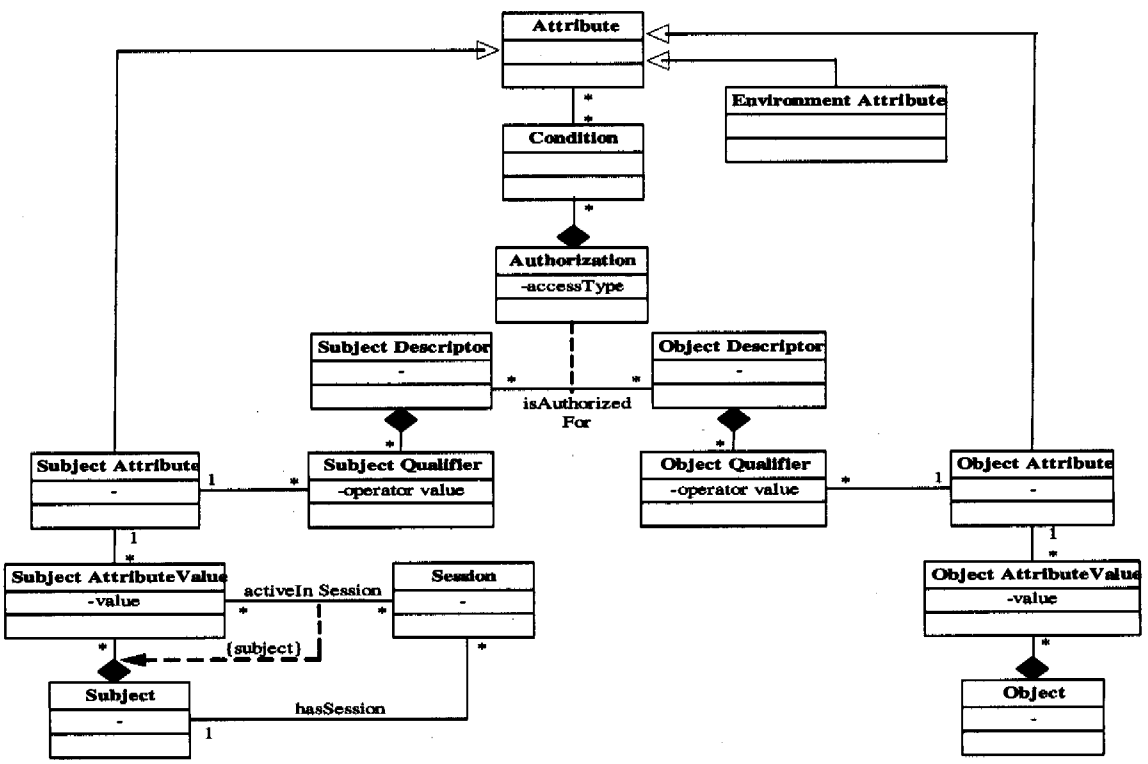

图 2 扩展 ABAC 模型图

2.2 扩展的 ABAC 模型

在基本的 ABAC 模型中，授权主要基于主体属性和客体属性，没有考虑与环境相关的属性。在扩展的 ABAC 模型（图 2 所示）中，我们引进了环境属性，与主体属性和客体属性

一起，用于授权决策中。另外，我们还增加了会话(session)的概念。利用会话概念，可以实现"最小权限"规则，也就是说主体只能获得它完成当前任务所需的权限。因为类似于 RBAC 模型，在一次会话中，主体只能激活分配给的属性的一个子集，只有被激活的属性才能用于访问控制。除此之外，利用会话概念，还可保护用户隐私。因为用户可根据完成任务的需要，只提供相应的属性，而不提供完成任务中不需要的其他属性。

3 ABAC 与 RBAC 的比较

因为基于角色的访问控制(RBAC)[5]减少了管理费用，增强了管理灵活性，所以近来被广泛重视。许多研究者提出将 RBAC 用于 Web 服务的访问控制之中，如 Rafae Bhatti 等人将 RBAC 与 XML 结合，提出了可用于 Web 服务的访问控制的 X-RBAC[6]模型；Roosdiana Wonohoesodo[7]提出了基于扩展的 RBAC 的 Web 服务访问控制模型。为了适应 Web 服务的动态性，考虑访问控制的上下文，Rafae Bhatti[8]提出了基于上下文与 RBAC 相结合的 Web 服务访问控制模型。但由于角色是静态的，角色通常与用户身份相关联，角色分配由人工实施。当用户数量较大时，管理工作量仍然很大；特别是当服务请求者和资源在不同的安全域时，管理规模是一个严重的问题。另外，在 Web 服务环境中，用户身份通常是事先不可知晓的，且是动态变化的，基于用户身份或基于用户角色的访问控制，不太适合于 Web 服务的动态异构环境，至少需要进行适当的扩展。与 RBAC 相比，ABAC 可提供基于上下文的控制决策等诸多优点。下面我们从认证与授权的耦合度、是否支持基于上下文的授权、授权决策的过程的性质和授权层次等方面，对 ABAC 与 RBAC 的进行比较。

(1)认证与授权的耦合度。对 ABAC，认证与授权功能上是独立的，通常是不同的服务，符合 AAI 框架的要求。对 RBAC，认证与授权是紧耦合的，通常是一个功能模块，不符合 AAI 框架的要求。

(2)支持基于上下文的授权。对 ABAC，它支持基于上下文的授权；访问规则不仅需要主体、客体满足相关的属性，授权还基于相关的环境上下文(环境属性)。对 RBAC，授权通常基于用户身份，不包括上下文(如环境)属性。

(3)授权决策的过程。对 ABAC，采用自动化的一致的授权决策；人工维护安全规则和属性，但机器进行自动化的授权决策。其中权限不是授予每个用户，而是通过属性的当前值和为被保护资源设定的规则动态地产生，决策过程由计算机实施。对 RBAC，采用人工的非一致的授权决策；安全管理员事先人工评估安全策略，并将结果作为权限保存，即人工的授权决策。

(4)授权层次。对 ABAC，采用基于资源层的授权；为资源制定的规则集定义授权所需的属性，决定用户的访问权；资源掌控相应的操作和规则。对 RBAC，采用基于角色或用户层次的授权；用户拥有访问权限；用户掌控相应的操作和任务。

从上可知，ABAC 和 RBAC 有很大的区别，但它们并不互相排斥。事实上，ABAC 能够完全支持所有的角色，一个角色仅仅是一种属性。ABAC 完全与 RBAC 兼容，并且增加了授权的粒度，它是 RBAC 的扩展。

4 ABAC 的使用探讨

要将 ABAC 运用于 Web 服务的访问控制中，首先要解决两个基本的问题，一是属性以何种方式来表示，二是如何来获取授权决策所需的属性。本节先探讨这两个问题，然后提出一种基于 XACML 的实施框架。

4.1 属性的表示

目前主要用 SAML(属性)声明和 X.509 属性证书这两种形式来表示属性。

4.1.1 SAML 声明

安全声明标记语言(SAML：Security Assertion Markup Language)[9]是 OASIS 组织发布的一种实现 Web 服务安全产品之间互操作的基于 XML 的标准，用于在业务伙伴之间安全地交换认证和授权信息。SAML 通过称之为"声明(assertion)"的消息在两个联邦域之间提供对登录和授权信息进行确认的标准化方式。一个域作为源站点认证基于浏览器的客户，而另一个则作为控制访问请求资源的目的站点。SAML 定义了认证声明、授权决策声明和属性声明等 3 种声明。声明可以根据主体、主体的属性、授权等信息来转换计算出(安全)验证信息(如该主体是否被允许访问某资源等)。声明被表达为 XML 格式，并具有嵌套的结构。单个声明可以包含若干不同的节点用于记录验证、授权、属性信息的数据。其中，属性声明声称特定主体具有特定的属性，可以通过 AttributeQuery 的方式从 Attribute Authority 处获取所需的属性，用于基于属性的访问控制决策中。另外，SAML 还定义了一个客户端向 SAML 安全验证方发送声明请求以及响应(Request/Response)的协议。这个协议包括对基于 XML 的请求/回应消息格式的定义。这些消息可以被绑定到很多主流的交换传送协议上。现在 SAML 只实现了与简单对象访问协议(SOAP：Simple Object Access Protocol)[10]的绑定，通过它可以实现基于 HTTP 的安全验证请求和响应。

4.1.2 X.509 属性证书

X.509 属性证书(AC：Attribute Certificat)[11]是属性权威机构(AA：Attribute Authority)签发的包含某持有者的属性集(如角色、访问权限、组成员等)和一些与持有者相关的信息的数据结构。由于这些属性集能够用于定义系统中用户的权限，因此作为一种授权机制的属性证书可被看作是权限信息的载体。AC 定义了一种安全提供授权决策信息的机制，实现了用户(身份)与其享有的权限属性的绑定。属性权威 AA 的数字签名保证了这种绑定的有效性和合法性。

由于 AC 不含用户的公钥，使用时要和公钥证书(PKC：Public-Key Certificate)结合使用，并且 AC 的有效期通常比 PKC 的有效期更短些。为了适应 Web 服务的 XML 架构和 XML 数字签名的需求，我们以基于 XML 的格式对 X.509AC 进行了扩展，称之为 XML-X.509AC。XML-X.509AC 的核心是属性集与证书持有者的结合，提供了属性与身份的绑定。同样地，XML-X.509AC 也包含序列号、发行机构、有效期等控制数据和数字签名。一个 XML-X.509AC 由两个嵌套的元素 AttributeCertificateInfo 和 Signature 组成，前者描述传输的信息，后者传输签名。在证书所有的域中，要特别提到 HolderBinding 和 Attributes 元素，前者用于主体 PKC 与其属性的绑定，而后者用于描述主体的标准属性信息和与应用相关的其他属性。下面，我们给出一个顾客购买飞机票的证书例子，其中顾客年龄为 50 岁，是大学"hust"的一名教授。

```
〈ac:AttributeCertificate〉
 xmlns:ds="http://www.w3.org/2000/09/xmldsig"
  xmlns:ac="http://www.hust.com/08/2003"
   〈ac:AttributeCertificateInfo
Id="Personal Attributes"Version="v2"〉
```

```
〈ac:HolderBinding〉
 〈ac:PKCX509IssuerSerial〉
  〈X509IssuerName〉
   C=US,O=RSA Data Security, Inc.,
   OU=Certification Authority
  〈/X509IssuerName〉
〈X509 SerialNumber〉7355476798〈/X509SerialNumber〉
  〈/ac:PKCX509IssuerSerial〉
  〈/ac: HolderBinding〉
〈ac:ACX509IssuerSerial〉〈/ac:ACX509IssuerSerial〉
〈ac:ValidityPeriod〉
〈NotBeforeTime〉2004-12-31T12:00:00
  〈/NotBeforeTime〉
〈NoAfterTime〉2005-10-30t12:00:00〈/NoAfterTime〉
〈/ac:ValidityPeriod〉
〈ac:Attributes〉
 〈ac:GenericAttribute Name="Age"
AttributeNameSpace="http://www.hust.com"〉
  〈AttributeValue〉50〈/AttributeValue〉
 〈/ac:GenericAttribute〉
 〈ac:GenericAttribute Name="University"
AttributeNameSpace="http://www.hust.com"〉
  〈AttributeValue〉hust〈/AttributeValue〉
 〈/ac:GenericAttribute〉
〈/ac:Attributes〉
〈/ac:AttributeCertificateInfo〉
〈ds:Signature〉…〈/ds:Signature〉
〈/ac:AttributeCertificate〉
```

4.2 属性的获取

客体属性和环境属性可由相应的提供者(安全管理员)直接定义,但主体属性通常维护在一个特殊的数据库中,或是通过属性证书或 SAML 声明分配给主体。当主体属性存储在客户端或可信任的第三方(如 AA)中时,可用如下的 3 种方式获取授权决策所需的属性信息,提供给策略决策点(PDP:Policy Decision Point)使用。

方法一:客户端发送属性。即客户端在请求访问某个服务时,随同请求一起发送属性给 PDP。这通常是通过将属性证书或 SAML 声明嵌入到 SOAP 消息的头部来实现的。

方法二:资源端请求属性。初始化客户端的访问请求后,资源端服务器请求客户端发送访问控制决策所需要的特殊属性,这需要双方的交互。

方法三:自动信任协商。这种方法通常在资源端不知晓请求方身份的情况下使用,也是目前重点研究的 Web 服务环境下获取属性的主要方式。它需要利用信任协商协议[12](如双方交换数字凭证)建立双方的信任关系,访问控制策略通常定义了为获取访问特殊的资源双方必须提供哪些凭证。

4.3 基于 XACML 的 ABAC 访问控制模型

扩展访问控制标记语言(XACML:eXtensible Access Control Markup Language)[13]是基于 XML 的标准,它定义了一种通用的用于保护资源的策略语言和一种访问决策语言。尽管 XACML 也有与其它策略描述语言同样的要素,如访问目标、主体(访问者)、操作和规则,但与以往的策略语言相比,XACML 是基于 XML 标准的,具有能够同时被人和计算机识别的特点,并且提供了一个标准化的访问控制决策模型;其资源访问策略是基于主体、资源和环境的属性,而不是基于请求者的身份,所以可提供比基于身份的、简单地拒绝访问或授权访问更细粒度的控制访问机制;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。基于 XACML 的 ABAC 访问控制模型如图 3 所示。

在基于 XACML 的访问控制体系结构中,当利用主体、资源和环境的属性进行授权决策时,XACML 提供了两种机制来解析来自请求或其它资源中的属性值:AttributeDesignator 和 AttributeSelector。AttributeDesignator 让策略以一个名称和类型来定义属性,同时可以提供一个发布者(issuer)可选项。然后 PDP 就在请求中寻找该属性的值,或判断该属性值是否在请求中可以被找到。AttributeDesignator 元素用于根据指定属性的名字、类型和发布者而从请求中获取属性值。AttributeSelectors 使得一个策略根据 XPath query 的形式查询一个属性值。只要提供一个数据类型和 XPath 表达式,就可以解析请求文档中的属性值。用它来比较请求与策略目标中的属性值,并最终产生访问控制结果。

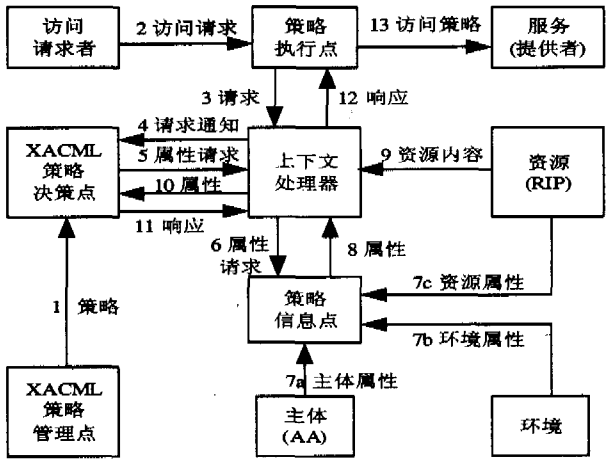

图 3 基本 XACML 的 ABAC 访问控制模型

在图 3 所示的 XACML 决策执行过程中,策略信息点到上下文处理器之间的属性请求与响应一般是 SAML 格式的请求与响应,而在上下文处理器与策略决策点之间的请求与响应是 XACML 格式的请求与响应。SAML 格式的请求与响应和 XACML 格式的请求与响应格式有所不同,由于一个 SAML AuthzDecisionQuery 不能够传递 XACML PDP 能够接收的、作为其请求上下文部分的所有信息,同样 SAML AuthzDecisionStatement 也不能传递包含在 XACML 响应上下文中的所有信息,因此需要上下文处理器在其中起相互转化的作用。上下文处理器可以用 XSLT(eXtensible Stylesheet Language Transformation)将 SAML 格式映射成 XACML 格式,反之亦然。XSLT 是一种用来转换 XML 文档结构的语言。

结束语 Web 服务的安全问题是制约其发展的关键因素,而有效的访问控制机制是其发展的重要保证。基于属性的访问控制是一种有效的具备诸多特点的控制访问机制,本文对其进行了建模,并讨论了其使用和实现问题。在将来的工作中,我们将重点研究如何利用自动信任协商机制来获取各种实体的属性,使系统成为动态的自适应的系统。

参 考 文 献

1 许峰,林果园,黄皓. Web Services 的访问控制研究综述. 计算机科学, 2005, 32(2):1~4

2 Bonatt P, Samarati P. A Unified Framework for Regulating Access and Information Release on the Web. Journal of Computer Security, 2002, 10(3):241~272

3 Li N, Mitchell J C. RT: A Role-based Trust-management Framework. In:Proceedings of the DARPA Information Survivability Conference and Exposition (DISCEX), Washington, D C, April 2003

4 Priebe T, Fernandez E B, Mehlau J I, et al. A Pattern System for Access Control. In: Proc. 18th Annual IFIP WG 11.3 Working Conference on Data and Application Security, Sitges, Spain, July

2004

5 Ferraiolo D F, Sandhu R, Gavrila S, et al. Proposed NIST standard for role-based access Control. ACM Transactions on Information and System Security (TISSEC), 2001, 4(3)

6 Bhatti R, Joshi J B D, Bertino E. Access Control in Dynamic XML-based Web-Services with X-RBAC. In: Proceedings of the First International Conference on Web Services, Las Vegas, USA, 2003

7 Wonohoesodo R, Tari Z. Role Based Access Control System for Web Services. In: Proceedings of the 2004 IEEE International Conference on Services Computing (SCC'04), Shanghai, China, 2004. 49～56

8 Bhatti R, Bertino E, Ghafoor A. A Trust-based Context-Aware Access Control Model for Web Services. In: Proceedings of the IEEE International Conference on Web Services (ICWS'04), San Diego, California, USA, 2004

9 OASIS Standard. Security Assertion Markup Language (SAML) V1. 1, October, 2003. http://www. oasis-open. org/committees/security/docs/cs-sstc-core-01. pdf

10 Simple Object Access Protocol (SOAP) V1. 1. May, 2000. http://www. w3. org/TR/2000/NOTE-SOAP-20000508

11 Farrel S, Housley R. An Internet Attribute Certificate Profile Authorization. http://www. itef. org/ rtf/rfc3281. txt, April 2002

12 Winsborough W H, Jacobs J. Automated Trust Negotiation in Attribute-based Access Control. In: Proceedings of the DARPA Information Survivability Conference and Exposition (DISCEX), Washington, D C, April 2003

13 OASIS Standard. eXtensible Access Control Markup Language (XACML) Version 1. 0. February, 2003. http://www. oasis-open. org/committees/xacml

(上接第 70 页)

个端点有一个单独的优化上传，不管现在的下载率如何处，它都不会阻塞。优化上传的端点，是以每三个阻塞检查周期(每周期 30 秒)循环的。30 秒对上传和下载操作足够了。

4)反冷落上传：有时某个端点服务器会被以前所有它能够下载的端点服务器阻塞。这种情况下，它只能得到较差的端点下载率，除非优化上传发现更好的端点。为了解决这个问题，如果来自某个特殊的端点单独块，在一分钟内没有响应，文件分块复制法认为它被该端点冷落了，而且除非作为优化上传，否则不会对之上传，因而对该端点作一个反冷落上传。这就导致多个同时的特殊上传(前面所提出的一个优化上传原则的特殊的反例)，将导致在系统颠簸时复制率尽快恢复。

5)仅仅上传：一旦某个端点完成了复制，它不再有可用的复制率决定那个端点去上传，现在它将变成更好的上传端点，专门用作上传，这对提高系统效率非常有益。

5.5 现实世界的经验

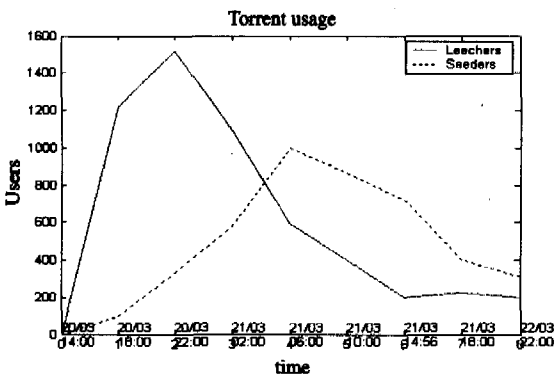

图 6 复制情况分析

这是一个大于 400 兆的文件在网格系统中提供复制的全过程，如图 6 所示，完成复制的数量(seeders 蓝线)和未完成复制的数量(leechers 红线)。其间，未完成复制的数量在文件可利用之后增加的很快，一旦达到顶点后将以指数的速度落下。相对而言，已完成复制的数量增长较慢，其峰值晚于未完成的峰值，之后缓慢下降，积分值大。未完成复制的数量成指数的剧烈增减和已完成复制的数量的稳定说明 LDAP 目录服务器的动态刷新与复制过程迅速扩展并完成，这证明文件分块复制法能进一步大幅度提高以 LDAP 目录分布式数据库为基础与核心的网格信息服务系统的效率。

本文中的文件分块复制法是源于因特网上流行的下载工具 Bittorrent。Bittorrent 不仅已经被使用，而且流传得很广，它为上百兆的文件下载服务，可面向上千个同时的下载者。

结论和以后的工作 网格技术使得广泛的大规模共享成为可能。网格资源信息服务是网格项目中的基础部分。文中我们根据网格资源信息的特点提出了由高度分布式的信息提供者和集合目录组成的网格信息服务基本框架，并且分析了它的基础 LDAP 协议，指出 LDAP 目录本质是一种分布式的数据库。由于网格信息系统分布广，容错性强，动态多样性，LDAP 目录信息树的动态刷新与复制将频繁发生。本文经过试验比较了多种复制策略，最终证明环形扩展和线形扩展的策略可大幅度提高系统效率；不仅如此，还用文件分块复制法，提出了技术框架和上传算法，把 LDAP 数据库文件分成若干块在多个端点间并行复制，这样进一步提高以 LDAP 目录分布式数据库为基础与核心的网格信息服务系统的并行效率。

目前我们研究了如何在节点级提高复制与刷新效率的策略，及节点级以下如何提高复制与刷新效率的策略；此外还应考虑 LDAP 目录的负载平衡问题，在什么地方复制将带来最佳的系统效果，等等。

参 考 文 献

1 Czajkowskiy K, Fitzgeraldz S, Fosterx I, et al. Grid Information Services for Distributed Resource Sharing. In: Proc. 10th IEEE International Symposium on High-Performance Distributed Computing (HPDC-10), IEEE Press, 2002

2 Foster I, Kesselman C, Tuecke S. The anatomy of the Grid: Enabling scalable virtual organizations. Intl. Journal of Supercomputing Applications, (to appear) 2002. http://www. globus. org/research/papers/anatomy. pdf

3 Howes T A. Lightweight Directory Access Protocol. http://www. kingsmountain. com/directory/doc/ldap/ldap. html

4 He Yanxiang, Fan Qianfeng, Zhang Lifei. Design of dynamic replication strategies for a grid computing . Computer Engineering, 2004(2)

5 Cohen B. Incentives Build Robustness in BitTorrent 2003. 5. http://www. bittorrent. com/bittorrentecon. pdf