基于角色和上下文的动态网格访问控制研究*)

姚寒冰　胡和平　卢正鼎　李瑞轩
(华中科技大学计算机学院　武汉 430074)

摘　要　网格计算旨在使地理上分散的资源实现全面共享与协同工作，网格环境的异构、动态和多域的特点为网格的安全研究带来了新的挑战。近年来在网格访问控制方面做了大量研究，大多在一个相对静态的假设下，主要依靠主体的标识来实现访问控制，缺少基于上下文的访问控制来适合动态的网格环境。文章提出了一个基于角色和上下文的动态网格访问控制(RCBAC)模型，RCBAC 扩展了 RBAC 模型，增加了上下文约束。RCBAC 从网格应用环境中获取与安全相关的上下文信息来动态地改变用户的权限，同时保留了传统 RBAC 模型的优点，这一访问控制模型正在实践中实施。

关键词　访问控制，网格安全，上下文，角色

Dynamic Role and Context-Based Access Control for Grid Applications

YAO Han-Bing　HU He-Ping　LU Zheng-Ding　LI Rui-Xuan
(College of Computer Science, Huazhong University of Science and Technology, Wuhan 430074)

Abstract　Grid computing is concerned with the sharing and coordinated use of diverse resources in distributed "virtual organizations". The heterogeneous, dynamic and multi-domain nature of these environments introduces challenging security issues. Despite the recent advances in access control approach applicable to grid computing, there remain issues that impede the development of effective access control models for Grid applications. Amongst them are the lack of context-aware models for access control, and reliance on identity or capability-based access control schemes. In this paper, we present an access control scheme that resolve these issues, and propose a Dynamic Role and Context-Based Access Control (RCBAC) framework which extends the RBAC with context constraints. The RCABC mechanisms dynamically grant and adapt permissions to users based on a set of contextual information collected from the system and user's environments, while retaining the advantages of RBAC model. RCBAC model is being implemented in our experiment.

Keywords　Access control, Grid security, Context-aware, RBAC

1　引言

网格使地理上分散的各种资源整合成一台超级计算机，从而实现资源共享与协同工作[1]。网格代表着一种崭新的资源共享和协同工作模型，已经成为国内外研究的热点。网格可以跨越多企业、多系统，从而使网格环境具有异构、动态和多域的特点，这为网格的安全研究带来新的挑战，使得其安全问题不同于通常意义上的分布式安全问题。各网格节点对安全控制的需求和采用的安全策略可能完全不同，因此网格的安全控制机制应该是通用的，全局的安全策略需要与本地的安全策略协调和交互，既要满足全局控制的需要，又可以满足用户自主控制的需求。网格的这些特点决定了其安全控制比一般的信息系统复杂得多。

近年来网格研究者在网格安全方面做了大量研究，提出了网格安全基础设施 GSI[2,3]，来满足在网格环境的安全认证与安全通信等安全需求。GSI 作为网格的安全标准，被广泛应用到网格环境中。但 GSI 并不能充分解决授权和访问控制的问题，Kesselman 和 Foster 等人提出了面向 VO 内用户管理的 CAS[4]，其核心思想就是为 VO 建立一个安全中心，用户与网格的交互过程从与 CAS 服务器的交互开始，CAS 服务器通过代理证书管理 VO 内所有实体的授权信息。与 CAS 相似的是 VOMS 技术，利用 LDAP 目录来保存 VO 内所有用户所属的组别及其权限[5]。另一类面向资源的管理技术如 Akenti[6] 和 PERMIS[7] 都不仅仅是针对网格提出的，考虑的是如何在分布式环境下实现对资源的有效访问控制，但对在网格中如何实现对资源的管理有借鉴意义。

上面列举的研究工作解决了网格的授权与访问控制的一些重要问题，但这些研究是在一个相对静态的假设下，主要依靠主体的标识来实现访问控制，不能解决在高度动态的网格环境中，主体的访问能力不仅依赖其标识，也和主体所处的与安全相关的上下文环境有关，这些上下文包括时间、位置、网络状况等信息。例如，考虑一个移动的网格用户，他的权限依赖于身份标识，同时与他所处位置(安全连接或不安全连接)、他要访问资源的状态(如负载)等有关。特别是他的权限可能随着上下文环境而改变。同样，一个网格服务与其它网格服务交互时，该服务的权限依赖其凭证和上下文环境。这些上下文参数能动态改变访问网格的权限，对访问结果有重要的影响。为了满足网格的动态访问控制的需求，必须扩展传统

*)本课题得到国家自然科学基金(60403027)资助。**姚寒冰**　博士研究生，主要研究方向为分布式系统安全、网格计算；**胡和平**　教授，主要研究方向为软件工程、数据挖掘、网络安全等；**卢正鼎**　教授、博士生导师，主要研究方向为软件工程、工程数据库、系统集成；**李瑞轩**　副教授，主要研究方向为系统集成、分布式系统安全、Web 数据管理、语义网、对等计算、边缘计算。

的访问控制技术，加入上下文约束。本文提出了基于角色和上下文的网格访问控制模型，第 2 节描述了网格访问控制的关键问题，第 3 节中提出了一个基于角色和上下文的动态网格访问控制模型(RCBAC)来解决网格的访问控制问题，在第 4 节中描述了 RCBAC 模型在网格中的实施框架，最后引出了本文的结论。

2 网格访问控制的关键问题

网格环境具有异构、动态和多域的特点，是由安全策略不同的多个管理域组成的异构系统，在网格访问控制的研究中有以下几个方面的关键问题：

通用性。在网格中，各个组织根据自己的需求定制访问控制策略，使网格的访问控制策略具有多样性的特点，而网格的访问控制机制需要支持这些根据不同需求自定义的访问控制策略。

自主控制。自主控制是指各个组织都可以独立地制定本管理域的访问控制策略，而不受其它组织访问控制策略和网格结构的影响。

动态适应性。由于网格的动态特点，网格的访问控制应该能反映网格的这种动态变化特点，能根据网格中与安全相关的环境信息对角色的权限做出动态的改变。特别是随着网格规模的扩大，移动设备也能加入网格应用中，网格的访问控制机制必须能满足这种动态改变角色权限的需求。

互操作。网格的访问控制需要提供跨组织的授权机制，以实现全局一体化的访问控制。这需要在各个管理域间进行访问控制的互操作，即跨管理域的授权。在跨管理域的授权中，由于访问的主体和客体可能属于访问控制策略不同的管理域，访问控制可能会发生冲突，这就需要一定的协调机制。

3 基于角色和上下文的访问控制(RCBAC)

3.1 RBAC 模型

RBAC 模型[8]的基本组件包括用户(USERS)、角色(ROLES)、客体(OBS)、操作(OPS)、权限(PRMS)和会话(SESSIONS)。用户与角色、角色与权限之间都是多对多的关系，即多个角色可以赋予同一个用户，多个用户也可以具有同一个角色；角色与权限之间也是一样。每个合法用户进入系统得到自己的控制的时候，就得到了一个会话，一个会话可以激活该用户全部角色的一个子集，用户能够获得全部被激活角色的所有权限。

RBAC 还包括层次 RBAC 和约束 RBAC，层次 RBAC 描述角色之间的层次关系，反映现实中职权之间的线性关系，实现多级安全系统中保密级别的要求。约束 RBAC 确定对角色分配和在会话中激活角色的约束条件，反映了现实中对用户在职责分离、赋予角色的前提和数量等方面的约束，主要包括静态职责分离(SSD)和动态职责分离(DSD)。静态职责分离是指若干特定角色不能同时赋予同一用户的约束关系；动态职责分离是指某些特定角色不能在一个用户会话中同时激活的约束关系。SSD 和 DSD 是实现最小特权原则的保证，其形式化描述这里不再赘述。

3.2 安全相关的上下文(Security Context)

基于上下文的访问控制是近年来的一个研究方向[9,10]，上下文包括时间、地点等环境信息。在网格中引入上下文，把传统的网格计算环境转换成了有感知功能的环境。当上下文变化时，访问结果也应不同。实际安全系统也需要对周围的环境有感知的能力，如访问控制可能依赖于时间和特殊的环境。可以利用上下文和用户的身份来确定用户的权限，即一个用户只有具有一定的身份并且处于相应的上下文环境中才可以具有相应的权限。这个策略就有很大的灵活性，因为也可以仅仅根据身份，不需要对上下文情况进行验证。这就是传统的 RBAC 方法。

图 1 说明了如何在网格环境中进行基于上下文的访问控制的基本原理。网格环境规定一些初始安全策略，这些安全策略与网格运行环境相关，这些网格运行环境称为上下文，上下文根据上下文触发器的请求而改变。当满足系统所设定的条件时，则激发相应的上下文变量。只有当特定的上下文变量被激活后，才可以授权，所以授权是动态的。这就意味着现在不可访问的资源在以后的某个时间，可能因为上下文的变化而可以访问。

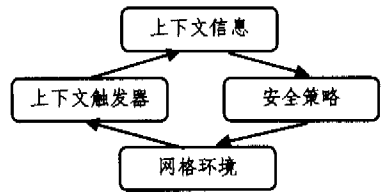

图 1 网格环境中的上下文访问控制

3.3 RCBAC 的定义

根据网格访问控制的特点，本文提出 RCBAC 模型。RCBAC 模型如图 2 所示，图中略去了 RBAC 模型的角色层次和角色约束。传统的 RBAC 模型可描述为：如果主体对某客体有访问操作请求，并且主体有访问操作权限，那么提供访问操作。它是一种被动安全模型，没有将执行操作所处的环境考虑在内，容易造成安全隐患。RCBAC 模型核心是采用“基于角色，结合上下文，在访问过程中提供动态实时的权限”。因此，在 RCBAC 中角色的访问权限并不是一成不变的，而是随着上下文的变化而变化。

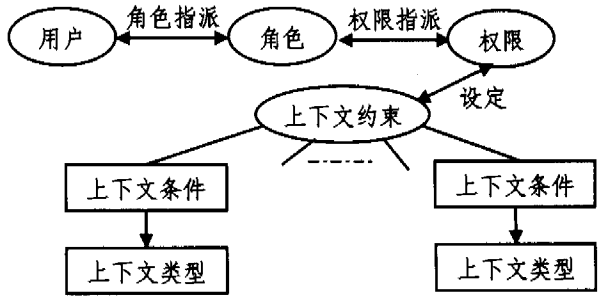

图 2 RCBAC 模型的核心思想

定义 1 上下文类型(CT)：与网格应用的访问控制有关的安全环境属性有一个名称属性 name，用来标识自己；还包含一个函数 *GetValue*()，用来在应用环境中取得 CT 的当前值。

CT 可能是时间、位置等一些具体的简单环境信息，也可能是抽象的概念，比如像网格信任管理中的信任级别。CT 的值是动态变化的，函数 GetValue()用来在应用环境中取得 CT 的当前值，它可能是一个本地函数，也可能是一个远程过程调用，在网格应用中一般用 Web service 实现。

定义 2 上下文集合(CS)：$CS = \{CT_1, CT_2, \cdots, CT_n\}$，$n \geqslant 0$，$\forall i, j, i \neq j, CT_i \neq CT_j$。

通过分析具体网格应用访问控制的要求，网格应用设计者可以决定需要的上下文类型集合。虽然网格应用的设计者在设计时决定了上下文集合，但网格的管理者可以根据需要在运行时动态地修改 CS，增加新的 CT 或从 CS 中删除不必要的 CT。

定义 3 上下文条件(CN)：关于 CT 的断言，CN=〈CT〉〈OP〉〈Value〉，CT∈CS，OP 可以是逻辑操作符{=，>，<，≥，≤，≠}，也可以是用户自己定义的操作符，Value 是管理员为 CT 设定的值。

CN 用来比较 CT 的当前值和设定的 CT 值，返回一个布尔型的值，满足条件返回真，否则为假。计算 CN 的过程成为上下文评估。

定义 4 上下文约束(CC)：一个规则表达式，基于这种形式的表达式，能描述复杂的安全要求。$CC = CL_1 \cup CL_2 \cup \cdots \cup CL_m$，$CL = CN_1 \cap CN_2 \cap \cdots \cap CN_n$。

假设网格应用的上下文集合 CS = {Time，IPAddress，TrustLevel}，有如下访问规则：域 DA 中的普通信任级别的用户在 8:00 和 18:00 之间可以访问网格资源，否则需要用户具有高的信任级别，这一规则的上下文约束表达式如下：CC=(Time>8:00∩Time<18:00∩IPAddress in DA∩TrustLevel=Normal)∪(TrustLevel≥High)。表达式中 in 是用户自定义的操作符。

定义 5 RCBAC 模型 = {USERS、ROLES、OBS、OPS、PRMS、SESSIONS、CC}

CC 是在 RBAC 模型中加入的上下文约束，其它元素都是 RBAC 模型中的基本元素，已经在 RBAC 模型中定义，这里不再重复定义。

定义 6 授权策略(AP)：一个三元组(R，P，C)，R∈ROLES，P∈PRMS，C∈CC。如果 CC 为空，该策略就转变为 RBAC 的授权策略。

定义 7 访问请求(AR)：一个三元组(R′，P′，RC)，R′∈ROLES，P′∈PRMS，RC 是 CS 集合中的元素运行时的值，即 RC={CT_1.GetValue()，CT_2.GetValue()，…，CT_n.GetValue()}。

对一个访问请求 AR，仅当存在一个 AP，R′∈R，P=P′，在 RC 这个条件下计算 AP 中 C 的值，如果计算过程返回值为真，AR 被授权访问，否则禁止访问。

3.4 上下文评估算法

根据前文的描述，给出如图 3 所示的上下文评估算法。该算法根据网格应用中定义的安全策略集合，从应用环境中取得上下文类型的当前值，决定一个访问请求 AP 能否访问请求的资源。

(1) 检索所有的与当前的访问请求(AR)有关的授权策略(AP)。

(2) 提取 AP 中的上下文约束(CC)，提取 CC 中的 CL，计算 CL 中的 CN，返回一个布尔值。

(3) 如果评估过程返回值为真，AR 被授权，否则拒绝访问。

```
Algorithm: RequestPermission (AccessRequest ar)
CPS = { } //初始化预选访问策略
for each AP in PS//PS 网格应用中的访问策略集合
    if (ar.R' ∈ AP.R) and (ar.P' = AP.P)
        put AP into PS
    end if
end for
result = false
for each AP in CPS
     if (EvaluateContexts(AP.C) is true)
        result = true
        break
     else
        result = false
     end if
end for
return result
```

```
Algorithm: EvaluateContexts(Constraint rc)
for each CL in rc
   for each CN in CL
       if (<CP.getvalue()><OP><VALUE> = false)
          //CP.getvalue 取得 CP 的运行时值
            //VALUE 是 CP 的设定值
          CL = false
          break
       end if
   end for
   if (CL = true)
       return true
   else
       continue
   end if
end for
return false
```

图 3 上下文评估算法

4 RCBAC 在网格中的实施框架

RCBAC 模型已在网格实验环境中实施，图 4 给出该实验网格访问控制框架。

认证服务：提供一些必要的服务来访问网格和网格中的资源。通过这个入口，用户能够无缝地对网格中的资源进行访问和操作。认证服务模块处理用户的登录，为用户创建代理证书和 Session。在用户通过认证后，与网格节点的访问控制服务联系，同时结合上下文 Agent 获取的上下文信息建立访问授权策略，保存在 Session 管理器中。

Session 管理器：保存用户的访问状态信息和授权信息，与 Web 中的 Session 类似。

上下文 Agent[11]：从网格应用环境中获取上下文信息。上下文改变时，上下文 Agent 改变 Session 管理器中的授权信息，保证访问权限随上下文而改变。

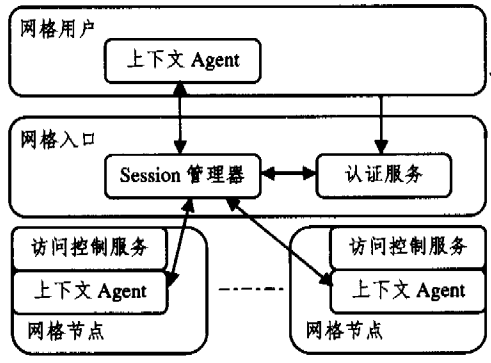

图 4 RCBAC 在网格应用中的实施框架

当一个网格用户通过网格入口访问网格系统中的资源时，如果通过了网格认证服务的认证，认证服务就会和相关的网格节点联系，网格节点的访问控制服务就会赋予用户一系列角色，同时认证服务在用户机器上建立上下文 Agent，用来获取与用户相关的上下文。同样，在网格节点上，建立上下文 Agent，用来获取共享资源的上下文。这些 Agent 能动态地从应用环境中获取上下文，把传统的网格计算环境转换成了有感知功能的环境。当上下文变化时，访问结果也应不同。

结论 为满足网格应用中复杂的访问控制需求，本文提出动态的基于角色和上下文的网格访问控制模型，该模型具有以下优点：①扩展了 RBAC 模型，加入了上下文约束，具有 RBAC 模型的优点。②该模型能动态地实施，具有很高的灵活性。在设计时，管理员能规定复杂的上下文敏感的授权策略。在运行时，基于动态变化的上下文，计算上下文约束条件，授权服务能自动实施上下文敏感的授权策略。③该模型易于扩展。上下文信息与应用程序分离，上下文的类型定义与访问控制规则无关，改变上下文的类型定义不会影响应用程序。

目前，RCBAC 模型已经在实验网格应用平台中作为核心的授权与访问控制模块。在下一步研究工作中拟讨论上下文约束的引入对 RBAC 模型的角色约束和角色层次的影响，从而以完整的基于角色和上下文的访问控制机制实现用户自定义的网格访问控制。

参 考 文 献

1 Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International Journal of High Performance Computing Applications, 2001, 15(3): 200～222

2 Foster I, Kesselman C, Tuecke G, et al. A Security Architecture for Computational Grids. In: ACM Conference on Computer and Communications Security Conference, 1998. 83～91

3 都志辉，陈渝，刘鹏. 网格计算[M]. 北京：清华大学出版社，2002. 67～80

4 Pearlman L, Welch V, Foster I, et al. A Community Authorization Service for Group Collaboration. In: Proceedings of the IEEE 3rd International Workshop on Policies for Distributed Systems and Networks, 2002

5 Alfieri R, Cecchini R, et al. VOMS, an Authorization System for Virtual Organizations. http://gridauth. infn. it/docs/VOMS-Santiago. pdf

6 Johnston W, Mudumbai S, Thompson M. Authorization and Attribute Certificates for Widely Distributed Access Control. In: Proceedings of IEEE 7th International Workshops on Enabling Technologies: Infrastructures for Collaborataive Enterprises, 1998

7 Chadwick D W, Otenko A. The PERMIS X. 509 role-based privilege management infrastructure. In: Proceedings of the Seventh ACM Symposium on Access Control Models and Technologies, Monterey, Califorinia, USA. 2002

8 Sandhu R, Coyne E, Feinstein H, et al. Role-Based Access Control Models. IEEE Computer, 1996, 29(2): 38～47

9 Mostéfaoui G K, Brézillon P. A Generic Framework for Context-Based Distributed Authorizations. In: Proc. 4th International and Interdisciplinary Conference on Modeling and Using Context (Context'03), LNAI 2680, 2003. 204～217

10 McDaniel P. On Context in Authorization Policy. In: Proceedings of the Eighth ACM Symposium on Access Control Models and Technologies, 2003

11 Deloach S A, Wood M F, Sparkman C H. Multiagent Systems. Int'l J Software Eng and Knowledge Eng, 2001, 11(3): 231～258

(上接第 21 页)

行了分类比较，重点介绍了不同缓存策略面临的问题、解决的思路和局限性。

流媒体代理缓存技术除了自身仍然面临着众多的选择和折衷，在其他很多方向上还有很多问题有待解决。

首先，随着无线网络的不断发展，针对无线信道传输可靠性差、误码率高、用户移动等特点，建立有效的缓存策略，使得部署在传统 Internet 与无线网络边缘的代理服务器为无线用户提供高效的流媒体服务，是目前研究的一个热点。

其次，现有的缓存策略的研究都是将 Web 对象与流媒体对象分开考虑，这与大多数实际情况不符，如何建立高效的混合缓存策略也是需要研究的问题。

最后，合作式缓存作为提高缓存系统扩展性的一个方案得到了广泛关注，设计一个流媒体对象在合作式缓存系统中进行高效分布和缓存的方案，对提高流媒体缓存技术的扩展性具有重要意义。

参 考 文 献

1 Miao Zhourong, Ortega A. Scalable proxy caching of video under storage constraints. IEEE JSAC, 2002, 20(7): 1315～1327

2 Zhang Z-L, Wang Y, Du D H C, et al. Video staging: A proxy-server-based approach to end-to-end video delivery over wide-area networks. IEEE Trans Networking, 2000, 8: 429～442

3 Dan A, Sitaram D. A generalized interval caching policy for mixed interactive and long video workloads. In: Proc. IS&T/SPIE Conf on Multimedia Computing and Networking (MMCN'96), 1996

4 Tewari R, Vin H, Dan A, et al. Resource-based caching for web servers. In: Proc. SPIE/ACM Conf on Multimedia Computing and Networking (MMCN'98), 1998

5 Sen S, Rexford J, Towsley D. Proxy prefix caching for multimedia streams. In: Proc. IEEE INFOCOM'99, 1999

6 Wu K, Yu P, Wolf J. Segment-based proxy caching of multimedia streams. In: Proc. 10th Int'l World Wide Web Conf. (WWW'01), 2001

7 Chen S, Shen B, Wee S, et al. Adaptive and lazy segmentation based proxy caching for streaming media delivery. In: Proc. NOSSDAV'03, 2003

8 Chen Songqing, Shen Bo, Wee Susie, et al. Investigating Performance Insights of Segments-based Proxy Caching of Streaming Media Strategies. In: Proceedings of SPIE/ACM International Conference on Multimedia Computing and Networking (MMCN'04), 2004

9 Tang Xueyan, Zhang Fan, Chanson S T. Streaming Media Caching Algorithms for Transcoding Proxies. In: Proc. ICPP 2002, 2002

10 Sasabe M, Taniguchi Y, Murata M, et al. Proxy Caching Mechanisms with Quality Adjustment for Video Streaming Services. IEICE Transactions on Communications Special Issue on Content Delivery Networks, 2003, E86-B: 1849～1858

11 Shen Bo, Lee Sung-Ju, Basu S. Caching Strategies in Transcoding-Enabled Proxy Systems for Streaming Media Distribution Networks. IEEE Trans On Multimedia, 2004, 6(2)

12 Vishwanath M, Chou P. An Efficient Algorithm for Hierarchical Compression of Video. In: Proc. IEEE Intl Conf Image Processing, 1994

13 Kangasharju J, Hartanto F, Reisslein M, et al. Distributing Layered Encoded Video through Caches. IEEE Trans On Computers, 2002, 51(6)

14 Rejaie R, Yu Haobo, Handley M, et al. Multimedia Proxy Caching Mechanism for Quality Adaptive Streaming Applications in the Internet. In: Proceedings of IEEE Infocom'2000, 2000

15 Podlipnig S, Boszormenyi L. Replacement Strategies for Quality Based Video Caching. In: IEEE International Conference on Multimedia and Expo (ICME), 2002, 2: 49～52

16 Zink M, Heckmann O, Schmitt J, et al. Polishing: a technique to reduce variations in cached layer-encoded video. In: Euromicro Conference, 2003. 249～254

17 Liu J, Chu X, Xu J. Proxy Cache Management for Fine-Grained Scalable Video Streaming. In: Proc. IEEE INFOCOM'04, 2004

18 Rejaie R, Kangasharju J. On Design and Performance Evaluation of Multimedia Proxy Caching Mechanisms for Heterogeneous Networks. In: Proceedings of IEEE International Conference on Multimedia and Expo, 2000

19 Radulovic I, Frossard P, Verscheure O. Adaptive Video Streaming in Lossy Networks: versions or layers? In: Proceeding of IEEE ICME, 2004

20 Yamada T, Wakamiya N, Murata M, et al. Implementation and evaluation of video-quality adjustment for heterogeneous video multicast. In: Proceedings of APCC, 2002. 454～457