그리드 사용자에 대한 정책기반 접근 제어 시스템 설계 및 구현

김경수° 김법균 황호전 곽의종 두길수 안동언 정성종
전북대학교
{boarder°, kyun, hjhwang}@duan.chonbuk.ac.kr, {kej, dgs, duan, sjchung}@moak.chonbuk.ac.kr

Design and Implementation of Policy based Access Control System for Grid user

Kyong Su Kim°, Beob Kyun Kim, Ho Jeon Hwang, Eu Jong Kwak, Gil Soo Doo, Dong Un ANN, Sung Jong Chung
Dept. of Computer Engineering, Chonbuk National University

a요 약

차세대 통신에서는 컴퓨터자원들이 지속적으로 대용량화 .고속화되는 추세이며 특히 생명공학, 유전공학, 유체역학 , 기상기후 예측 등 여러 과학 분야에서 단일자원으로는 제공하기 힘든 계산 및 저장자원을 요구 하고 있다. 이러한 문제를 해결하기 위해서 지리적으로 분산 되어있는 자원들을 연결하여 마치 단일 자원을 사용하는 것처럼 해주는 서비스인 그리드가 대두 되었다. 그러나 그리드 사용자가 작업을 수행시키기 위해서는 자신의 DN(Distinguished Name)을 Remote Machine상에 Local User Account를 바인딩 시켜줘야 한다. 따라서 각 사이트 관리자는 그리드 서비스를 제공하기 위해 수많은 그리드 사용자의 DN과 Local User Account를 바인딩 처리를 해 주어야 한다. 그러나 사실상 현실적으로 불가능하다. 이러한 문제를 해결하기 위해서 본 논문에서는 그리드 사용자에 대한 정책기반 접근 제어 시스템을 설계 및 구현했다.

1. 서 론

초고속 인터넷 통신망을 활용하여 지리적으로 분산되어 있는 유휴자원들을 통합하여 하나의 메타 컴퓨팅 환경을 제공하기 위한 연구가 활발히 이루어지고 있다. 대표적인 메타 컴퓨팅 환경으로 그리드 컴퓨팅 환경을 예로 들 수 있다. 그리드 컴퓨팅 환경은 광대역 통신망에 연결된 슈퍼컴퓨터, 클러스터, 워크스테이션, 대용량 저장장치와 같은 이기종간의 자원을 공유함으로써 대용량의 정보를 분석하고 처리할 수 있는 강력한 파워를 가진 가상의 컴퓨팅 환경이다. 따라서 그리드 컴퓨팅 환경은 안정적이고, 유동적이고, 강건해야 하며, 또한 그리드 컴퓨팅 환경에 참여하는 각 사이트의 자치권을 존중해주어야 한다.

각 사이트가 그리드 서비스를 제공하기 위해서는, 지리적으로 분산되어 있는 자원들을 단일 컴퓨팅 환경으로 통합할 수 있는 그리드 미들웨어 기술이 필요하다. 그리드 미들웨어로는 Globus[1], Legion, Condor 등이 그 대표적인 예인데, 현재 가장 많이 사용이 되고 있는 미들웨어가 Globus Toolkit이다. Globus Toolkit 미들웨어 상에서는 Single Sign-On을 통해 그리드상의 허용된 모든 자원들을 사용할 수 있게 된다.

Globus Toolkit 미들웨어를 기반으로 하는 그리드 컴퓨팅 환경에서는, 그리드 사용자가 작업을 수행시키기 위해서는 자신의 DN(Distinguished Name)[2]을 Remote Machine상에 Local User Account를 바인딩 시켜줘야 한다. 따라서 각 사이트 관리자는 그리드 서비스를 제공하기 위해 수많은 그리드 사용자의 DN과 Local User Account를 바인딩 처리를 해 주어야 한다. 그러나 사실상 현실적으로 불가능하다.

따라서 본 논문은 각 사이트의 관리자가 그리드 사용자를 위해 일정량의 Local User Account를 만들어두고 신뢰할 수 있는 그리드 사용자로부터 서비스 요청이 있을 시, 그리드 사용자의 DN과 Local User Account를 서비스가 수행되는 동안만 일시적으로 일대일 바인딩 시켜, 바인딩된 Local User Account로 그리드 사용자의 작업을 수행시키도록 하였다. 또한 각 사이트의 관리자는 국적, 지역, 소속기관에 따라 로컬 자원의 사용 권한을 결정할 수 있는 서비스 정책을 수립하여, 관리자의 정책에 따라 그리드 사용자의 요구사항을 충족시켜줄 수 있도록 하였다. 그럼으로써 그리드 컴퓨팅 환경에 참여하는 각 사이트의 자치성을 존중하고 올바른 정책 모델을 제시하여 그리드를 위한 정책 기반 접근 제어 시스템을 설계 및 구현하고자 한다.

2 장에서는 위에서 설명한 기능을 담당하는 그리드를 위한 정책 기반 접근 제어 시스템의 구조에 대해서 설명하고 , 3 장은 구현 , 4 장은 결론 및 향후과제에 대해서 설명한다.

2. 그리드를 위한 정책기반 접근 제어 시스템의 구조

2.1 전체 구조

그리드를 위한 정책기반 접근 제어 시스템의 구조는 크게 그리드 사용자 측에 위치하는 Client Access Controller 와 그리드 자원 제공자 측에 있는 Server Access Controller로 구성된다[그림 1]. 이 둘은 그리드

사용자의 DN 과 이용 가능한 Group List 를 서로 주고 받으면서 상호작용을 한다.

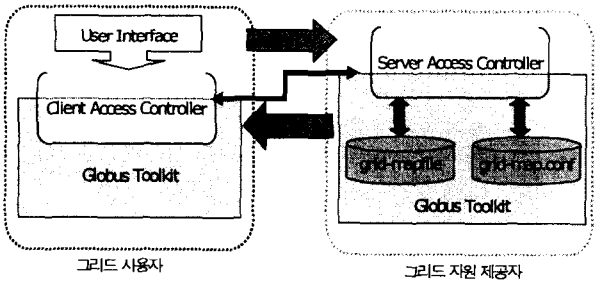

그림 1 .그리드 사용자에 대한 정책 기반 접근 제어 시스템 구조

2.2 각 Controller 에 대한 세부구조

2.2.1 Client Access Controller

*Client Access Controller*는 그리드 사용자의 사이트에 위치한다. 이 Controller의 대부분의 역할은 그리드 사용자의 요청을 받는 일을 한다. 예를들면 Server Access Controller와의 통신을 통하여 사용자의 Group List 요청을 하는 역할과 사용자로부터 선택된 Group List 에 대한 바인딩 요청을 한다. 또한 작업이 끝나면 바인딩 해제 요청도 이 Controller를 통하여 이루어진다.

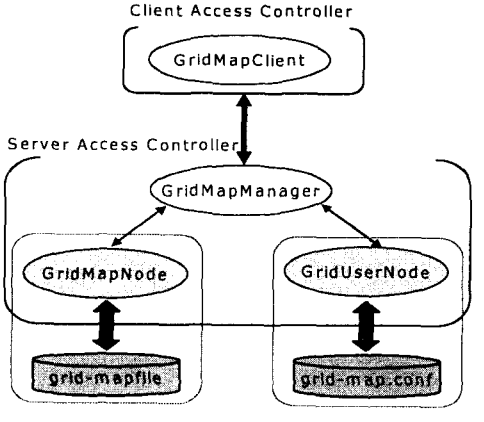

그림 2. 각 *Controller*에 대한 세부구조

2.2.2 Server Access Controller

Server Access Controller는 그리드 자원을 제공하는 각 사이트에 위치한다. 이 Controller는 크게 3부분으로 구성된다. 전체적인 기능을 총괄하는 GridMapManager 와 grid-mapfile 에 사용자의 DN과 Local User Account와 바이딩 역할을 해주는 GridMapNode가 있으며 마지막으로 자원 제공자의 정책을 적용한 *grid-map.conf* 파일을 관장하는 GridUserNode 로 구성된다. 이 Controller는 Client Access Controller로부터 요구되는 Group List 의 전송, Local User Account와의 바인딩 설정 및 해제 하는 일을 담당한다.

2.2.3 grid-map.conf

아래는 사이트의 정책에 맞는 사용자를 바인딩 시켜주는데 이용되는 *grid-map.conf* 파일의 예이다.[표1] 각 사이트의 그리드 자원 관리자의 정책에 맞게 파일을 수정 및 변경해서 사용한다.

```
# grid-map.conf file
[High_Level_Group] # Local Group Account
        Kyongsu    # Local User Account
        Jihong
[Middle_Level_User]
        Chuni
        Kyongick
[Low_Level_User]
        YoungJu
        Donggi
```

표 1 . *grid-map.conf* 파일

2.2.4 그리드 자원 사용권한 정책

각 그리드 자원 제공자의 관리자는 국가이름, 지역, 소속기관, 전공분야 , 제제조치기록 등[그림3]에 따라 로컬 자원의 사용 권한을 결정할 수 있는 서비스 정책을 수립한다. 그리드 사용자가 그리드 자원을 요청하게 되면 그리드 사용자의 User DN을 보고 그 사용자가 적법한 사용자인지 , 어느 정도의 자원을 허락하게 할 건지를 판단한 후 그리드 자원의 사용을 수락하거나 거절한다.

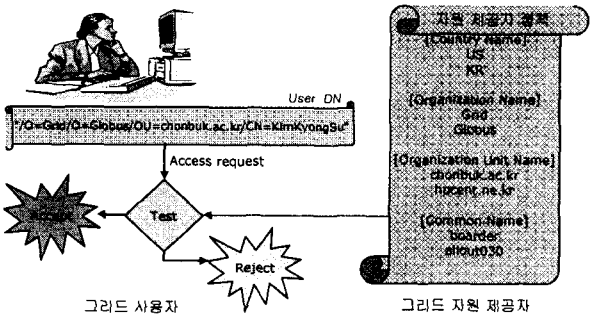

그림 3. 그리드 자원 사용권한 정책

이런 정책을 통하여 그리드 자원을 제공하는 제공자들은 서로 다른 사이트들의 특성에 맞는 자원 정책의 수립이 가능하며 효율적인 그리드 자원의 사용을 가능하게 한다.

2.4 그리드 사용자와 그리드 자원 제공자간의 흐름도

아래는 그리드 사용자와 그리드 자원 제공자 간에 일어나는 처리과정을 보여준다.[그림 4]

1) Client Access Controller를 통하여 Server Access Controller 에게 사용자의 DN을 제공하고 바인딩이 가능한 *Group List* 요구한다.

2) Server Access Controller는 바인딩 가능한 Group

List를 알려준다.
3) 사용자는 원하는 Group을 선택 한 후 Server Access Controller 에게 바인딩 해줄 것을 요구한다.
4) Server Access Controller 은 사용자의 DN을 선택한 Group으로 바인딩 시켜주며, 사용자에게 바인딩의 성공 여부를 전송한다.

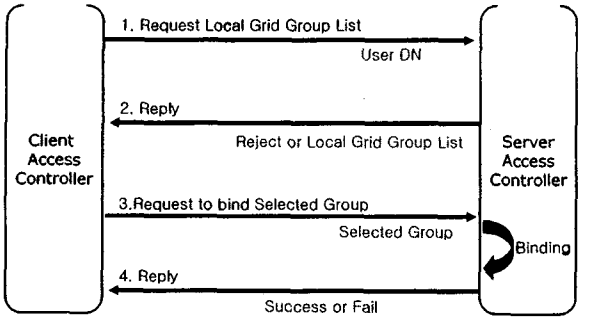

그림 4 . 그리드 사용자와 그리드 자원 제공자간의 흐름도

3. 구현

본 논문은 Globus Toolkit 2.0을 기반으로 하고 Java Cog Kits 과 Java 언어를 사용하여 구현 했다. 먼저 가장 널리 사용이 되고 있는 그리드 미들웨어인 Globus Toolkit을 사용했다. 둘째로 Globus 의 프로토콜과 기능을 자바로 작성한 라이브러리인 Java Cog Kits을 사용했다. 이 라이브러리를 이용하여 개발자나 사용자들은 그리드 자원 검색, 작업의 실행, 사용자 인증, 파일전송 등을 할 수 있다. 구현을 위해서 Java Cog Kits을 사용한 이유는 플랫폼에 독립적인 Java 언어로 만들어져 이기종의 환경 즉 그리드 환경에 적합하며, 컴포넌트 형식으로 만들어져 있기 때문에 쉽게 필요한 기능들을 추가 및 변경이 용이하기 때문이다. 셋째로 Java 의 AWT[9]를 이용하여 User Interface를 구현했으며 Client Access Controller 와 Server Access Controller와의 통신을 위해서 RMI[10]를 이용하였다.

아래의 구현 화면은 Job 실행부분에 관한 화면은 삭제하고 그리드 사용자에 대한 정책 기반 접근 제어 시스템의 구조와 관련 있는 부분만 언급했다.

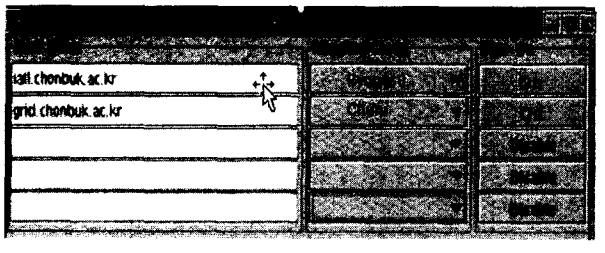

그림 6 . GUI Interface를 통한 작업 실행 결과

4. 결론 및 향후 과제

본 논문에서는 각 자원 제공자들의 정책을 반영한 그리드 사용자를 위한 정책 기반 접근 제어 시스템을 설계하고 구현 하였다. 그럼으로써 그리드 서비스를 제공하기 위해 수많은 그리드 사용자의 DN과 Local User Account를 바인딩 처리를 해 주어야하는 문제점을 해결하였고 , 또한 grid-map.conf 파일 과 자원 제공자 정책을 관리함으로써 서로 다른 사이트들의 정책을 최대한 반영한 시스템을 구축 할 수 있었다.

향후 추가 되어야 할 부분은 각 사이트의 관리자가 그리드 사용자를 위해 일정량의 Local User Account를 미리 만들어 두어야하는 번거로움을 해결해야 하며 나아가서는 점차 그리드 기술이 광범위 해지고 사용자들이 증가하기 때문에 각각의 그리드 사용자들에 관한 세부적인 어카운팅 정보[4]를 얻어내고 취합하고 그리드 사용자에게 자원 사용에 관한 과금을 부과하는 기능 등을 포함해야 할 것이다.

5. 참고 문헌 (또는 Reference)

[1] I. Foster, C.Kesselman and S.Tuecke, " The Anatomy of the Grid: Enabling Scalable Virtual Organizations," Journal of the International Supercomputer Applications , 2001.
[2] The physiology of the Grid: An open Grid services architecture for distributed systems integration, Ian Foster, Carl Kesselman, Jeffrey M.Nick, Steven Tuecke)
[3] G. von Laszewski, I.Foster, J.Gawor, W.Smith, and S. Tuecke, "Cog Kits: A Bridge Between Commodity Distributed Computing and High-Performance Grids," Proc. of the ACM Java Grande Conference, 2000.
[4]Thomas J.Hacker, Brain D.Athey Q.16,"Account Allocations on the Grid". Center for Parallel Computing University of Michigan. 2000.
[5] G. von Laszewski, J. Gawor, and P.Lane Java CoG Distribution http://www.globus.org/cog, Jan.2000.Version 0.8.6.
[6] G. C. Fox and W.Furmanski. Hpcc as High Performance Commodity Computing. http://www.npac.syr.edu/users/gcf/HPcc/HPcc.html,Dec. 1997
[7] I. Foster C. Kesselman(eds). Q.677,"The Grid: Blueprint for a New Computing Infrastructure" Mogan Kaufmann Publishers, 1998.
[8]http://www.globus.org/gram/gram+rsl_parameters.html
[9]http://java.sun.com/docs/books/tutorial/applet/practical/gui.html
[10]http://java.sun.com/docs/books/tutorial/rmi/index.html
[11] Globus Toolkit, http://www.globus.org
[12] http://www.ggf.org/5_ARCH/ACCT.htm
[13]http://www.gridforum.org