DOI: 10.3745/KIPSTC.2008.15-C.6.455

그리드 보안을 위한 역할 기반의 신뢰 협상 모델

조 현 숙[†] · 이 봉 환[††]

요 약

본 논문에서는 그리드 보안 프레임워크에서 디지털 인증서를 기반으로 신뢰를 구축하는 FAS(Federation Agent Server) 모델을 제안한다. 제안하는 FAS 모델은 기존의 RBAC(Role Based Access Control) 모델의 주요 요소인 사용자, 역할, 그리고 허가의 기본구조를 구체화하고 확장하여 연합 에이전트 서버를 설계함으로써 로컬정책에 따른 상세한 접근권한을 할당할 수 있는 시스템 독립적인 그리드 보안 통합 모델이다. FAS는 각 사용자가 어떤 역할을 가지는지를 결정하고, 역할에 따른 접근권한을 할당하며, 역할과 접근권한을 포함하는 속성 인증서를 발행하는 RDM, PCM 그리고 CCM의 세 가지 내부 모듈로 구성된다. RDM에서 신뢰 협상 과정을 통하여 사용자들이 서버의 정책을 확인하고 그들이 계획한 작업을 수행하는데 따르는 자신의 역할 및 접근권한을 선택할 수 있도록 설계함으로써, VO(Virtual Organization) 내의 모든 사용자들이 단일 사용자 계정으로 매핑하던 기존의 낮은 시스템 보안 레벨을 탈피하였다. PCM · CCM 과정을 통하여 어떤 작업들을 어떤 사용자가 수행할 수 있으며, 어떤 우선순위를 가지는지에 대해 제한하기 위해서 서로 다른 사용자 그룹과 역할에 대한 다양한 정책을 적용하고 인증서를 발행함으로써 보다 향상된 보안레벨을 가지고 그리드 서비스를 제공할 수 있는 기반을 마련하였다.

키워드 : 신뢰 관리, 신뢰 협상, 접근제어, 권한검증, 그리드 보안

RBAC-based Trust Negotiation Model for Grid Security

Hyun-Sug Cho[†] · Bong-Hwan Lee[††]

ABSTRACT

In this paper, we propose FAS model for establishing trust based on digital certificates in Grid security framework. The existing RBAC(Role Based Access Control) model is extended to provide permissions depending on the users' roles. The FAS model is designed for a system independent integrated Grid security by detailing and extending the fundamental architecture of user, role, and permission. FAS decides each user's role, allocates access right, and publishes attribute certificate. FAS is composed of three modules: RDM, PCM, and CCM. The RDM decides roles of the user during trust negotiation process and improves the existing low level Grid security in which every single user maps a single shared local name. Both PCM and CCM confirm the capability of the user based on various policies that can restrict priority of the different user groups and roles. We have analyzed the FAS strategy with the complexity of the policy graph-based strategy. In particular, we focused on the algorithm for constructing the policy graph. As a result, the total running time was significantly reduced.

Keywords : Trust Management, Trust Negotiation, Access Control, Authorization, Grid Security

1. 서 론

지금까지 디지털 세계에서의 신뢰 구축을 위한 접근법은 사용자 이름과 패스워드를 사용하는 아이디 기반의 접근법이 주류를 이루어 왔다. 그러나 인터넷과 같은 개방형 시스템에서는 상호 교류가 제3자들 사이에서 일어나기 때문에, 사용자들은 데이터 및 서비스를 제공받기 위하여 서로 다른 보안 도메인에서 접근하게 되었다. 개방형 시스템에서는 서비스의 공유와 인터랙션이 생산성과 효율을 높이는 중요한 요소로 자리매김하였으며, 이러한 서비스와 자원은 불법적인 접근으로부터 안전성을 제공받아야 한다. 기존의 아이디 기반의 접근법은 개방형 시스템에서는 불가능한 경우도 있고 불필요한 경우도 있다. 따라서 아이디를 대신할 수 있는 서비스 요구 주체의 역할 또는 속성을 근거로 접근하는 방법으로 사용자의 특정 속성(attribute) 및 역할(role)에 기반한 새로운 신뢰 모델이 요구되고 있다.

신뢰 협상[1-3]은 제3자들 사이에서 상대의 속성을 포함

※ "본 연구는 한국산업기술재단의 지역혁신인력양성사업 및 정보통신연구진흥원의 대학 IT 연구센터 지원사업 (IITA-2008-C1090-0801-0014)의 연구결과로 수행되었음".
† 정 회 원 : 대전대학교 교양학부 전임강사
†† 종신회원 : 대전대학교 정보통신공학과 교수(교신저자)
논문접수 : 2008년 4월 1일
수 정 일 : 1차 2008년 5월 30일, 2차 2008년 7월 21일, 3차 2008년 8월 18일
심사완료 : 2008년 9월 16일

하는 디지털 인증서를 교환함으로써 신뢰를 구축하는 새로운 접근법이다. 디지털 인증서의 속성에 기반하여 신뢰 문제를 해결하기 위한 시스템을 신뢰 관리시스템 (trust management system)[2]이라 한다. 여기서 신뢰는 반복적으로 인증서를 교환하고 인증서를 요청하는 신뢰 협상의 과정으로 구축된다.

신뢰 관리의 개념은 접근 제어와 많은 관련이 있다. 신뢰 관리 접근법은 "신뢰" 관계를 확장한 분산 접근 제어로 예를 들면, 신뢰 표현·신뢰 전파 그리고 신뢰와 직접적으로 연관된 보안 정책에 대한 개념이다. 여기서 접근 제어라는 용어의 사용은 인증과 권한 검증의 개념 모두를 포함한다. 인증(Authentication)이 주체를 구별하는 것과 관련이 있다면, 권한 검증(Authorization)은 한번 인증을 거친 주체가 접근 권한을 허가받는 것을 말한다.

현재 그리드 보안은 GSI(Grid Security Infrastructure)를 이용하여 인증, 권한검증 그리고 권한위임의 문제를 해결한다. 그러나 GSI에서 사용되는 그리드 맵 파일을 이용한 접근권한은 여러 가지 문제를 가지고 있다. 최근 그리드 상에서 접근권한의 문제는 그리드 보안 영역에서 가장 활발하게 연구되고 있는 분야로서, 그리드가 추구하는 두 가지 목표인 "가상의 슈퍼컴퓨팅 환경"을 구축하는 문제나 인터넷2와 같은 "하나의 차세대 인터넷 서비스"를 제공하는 문제 모두에서 중요한 해결 과제이다.

그리드 환경이 과학자들을 위한 미들웨어로서 의미가 있었던 과거와는 달리, IBM 등의 거대 벤더들이 그리드를 비즈니스 영역에서 수용하게 됨으로서 웹서비스 도메인으로 그리드가 통합되고 있는 실정이다. 그리드의 자원뿐만 아니라 보안까지도 웹서비스를 이용하려는 시도가 진행되고 있다. 그러나 기존의 다양한 보안 기술을 결합하여 그리드 보안 서비스를 시범적으로 도입하고 있으며, 그리드를 웹서비스로 통합하였을 경우에 대한 만족할 만한 보안 메커니즘은 부재한 상태이다. 이는 그리드 기술이 자원의 결합, 할당, 실행 등 다양한 작업을 위한 기반 기술을 갖추고 있음에도 불구하고 상용화되지 못하는 요인이 된다.

2. 연구의 배경

2.1 신뢰 협상

Matt Blaze와 그의 제자들이 처음으로 네트워크 서비스에 있어서 보안에 대한 구별 가능한 컴포넌트를 지시하는 말로 "신뢰 관리(trust management : TM)[4]"라는 용어를 사용하였다. 그들은 신뢰관리를 "보안 정책, 인증서 그리고 그들의 관계를 명세화하고 해석하기 위한 통합된 접근법"이라 정의하였다. 신뢰 관리 시스템의 핵심은 보안 정책과 인증서, 그리고 정책 컴플라이언스를 다루기 위한 일반적인 목적의 메커니즘의 집합이라는 것이다. 엔티티의 특성을 사용하여 신뢰를 구축하는 새로운 접근법이 "신뢰 협상(trust negotiation : TN)"이다. 하나의 TN은 공인받은 디지털 인증서를 반복적으로 노출하는 과정으로 구성된다. 이러한 인증서들은 상호 신뢰를 구축하기 위하여 소유자들의 특성을 입증해준다. 이와 같이 TN은 보안 정책과 인증서들을 공식화하는 개념을 다룬다. 즉, 신뢰 협상은 특정 인증서의 집합이 관련된 정책들을 만족하는지, 그렇지 않으면 제 3자로서 신뢰하는 것을 뒤로 미룰지를 결정하는 과정이다.

협상이 이루어지는 동안에 인증서 s의 노출은, 인증서 s를 노출하기 위해 만족해야 하는 필요조건을 가진 접근 제어 정책 p_s에 의해 유도된다. 일반적으로, 필요조건이란 인증서의 집합 $C' \subseteq C$ (단, C 는 모든 인증서의 집합)를 말한다. 각각의 인증서 $c_i \in C$ 에 대하여 각각의 정책 p_s는 다음과 같이 나타낼 수 있다.

$$p_s : s \leftarrow \Phi_s(x_1, \cdots x_k) \tag{1}$$

단, $p_s : s \leftarrow \Phi_s(x_1, \cdots x_k)$는 boolean 변수 x_i에 대한 표준식이 된다. Boolean 연산자는 $\vee$와 $\wedge$, 그리고 경우에 따라서 괄호가 필요하다. s는 p_s의 타겟으로 $\Phi_s(x_1, \cdots x_k)$는 p_s의 조건이다. 주어진 인증서의 집합 $C' \subseteq C$ 에 대하여, $g_{\Phi_s}(C') = 1$ 은

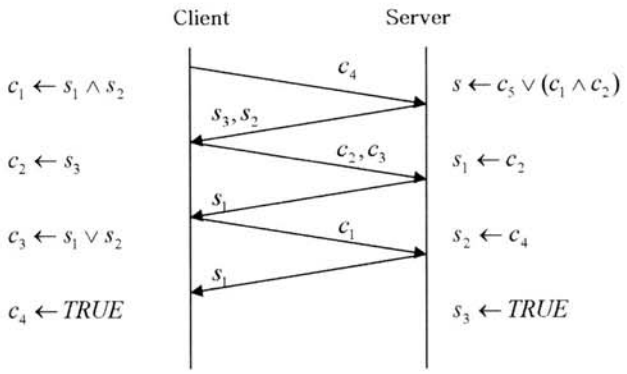
$$\Phi_s(x_1, \cdots x_k)\, x_i = 1 \Leftrightarrow c_i \in C' \tag{2}$$

으로 주어진 인증서의 집합이 조건을 만족한다는 것을 의미한다. 예를 들어, 만약에 $\Phi_s = (x_1 \wedge x_3) \vee x_2$ 라면 $g_{\Phi_s}(c_1, c_2, c_4) = 1$ 이고, $g_{\Phi_s}(c_1, c_4) = 0$ 이다. 그렇다면, 정책 p_s는 인증서의 집합

$$C' \in C \text{ if and } only \text{ if } g_{\Phi_s}(C') = 1 \tag{3}$$

을 만족한다. 협상이 이루어지는 동안 협상자는 상대로부터 전송받은 인증서의 집합이 C' 일 때, 인증서 s가 $g_{\Phi_s}(C') = 1$ 을 만족하면 협상을 종료하게 된다. 본 논문에서는 정책 내의 x_i를 c_i로 표기하기로 한다.

(그림 1)은 일반적인 신뢰협상 과정의 예를 나타낸다. (그림 1)에서 $p_s \leftarrow c_5 \vee (c_1 \wedge c_2)$이므로 $\Phi_s = c_5 \vee (c_1 \wedge c_2)$이다. 즉, 인증서 c_5 또는 c_1과 c_2를 전송받으면 Φ_s를 만족하게 되고, s를 노출하게 되는 것이다. 서버의 서비스 s를 클라이언트가 요청함으로써 신뢰협상 과정이 시작된다. (그림 1)에서 클라이언트의 접근 제어 정책은 왼쪽에, 서버의 접근 제어 정책은 오른쪽에 나타내었다. 서버와 클라이언트는 서로의 정책을 알지 못한 상태에서 협상을 시작한다. 서버로부터 c_4를 노출하라는 지시를 받지 않고, 클라이언트는 인증서 c_4를 노출한다. 그러면 서버는 어떤 전제 조건도 없는 s_3와 클라이언트가 보낸 인증서 c_4를 요구하는s_2를 전송한다. 그러면 클라이언트는 s_3와 s_2를 만족하는 인증서 c_2 및c_3를 전송하고, 서버는 c_2를 요구하는 s_1을 전송한다. 끝으로 클라이언트는 s_1을 만족하는 c_1을 전송한다. 결국 클라이언트는 서버의 서비스를 획득



(그림 1) 신뢰협상 과정

하기 위하여 필요한 인증서 $c_1 \wedge c_2$를 가지고 있어 $g_{\Phi_s}(c_1, c_2, c_3, c_4) = 1$을 만족하므로 서버는 클라이언트에게 서비스를 노출하게 된다. (그림 1)에서 서버의 서비스를 획득하기 위해 클라이언트는 자신의 모든 인증서를 노출해야만 한다.

인증서의 교환은 "전략(strategy)"이라고 불리는 각 협상자(negotiator)의 결정에 달려있다. 전략은 로컬 인증서와 로컬 정책, 상대 협상자로부터의 인증서에 대한 요구, 그리고 상대 협상자로부터 전송받은 인증서에 기초한다. 전략은 어떤 인증서가 노출되었고, 언제 협상이 종료하는지에 따라 조절된다[5]. 신뢰는 초기에 요구된 서비스가 허가되고 노출된 인증서에 대한 모든 정책이 만족되는 경우에 구축된다. (그림 1)의 경우 인증서 교환을 통하여 서비스를 획득하였으므로 성공한 협상(successful negotiation: $p_c : c \leftarrow TRUE$)이다. 반면에 협상에 실패할 경우, 즉 협상자가 요구하는 인증서를 가지고 있지 않을 경우 $p_c : c \leftarrow FALSE$ 로 실패한 협상(failed negotiation)이라 한다. 협상자들이 특정 전략을 적용하여 인증서 교환에 성공하였고, 이러한 인증서 교환 전략이 적용할 때마다 성공하는 전략을 "완벽한 전략(complete strategy)[6]"이라 부른다.

2.2 기존의 GSI의 문제점

그리드 기술에 웹 서비스 기술을 접목하려는 최근의 시도가 성공하기 위한 핵심이며 기본이 되는 필수적인 요구사항은 바로 보안이다. 글로버스에서 최근 발표된 GT4의 보안 모듈[7]은 SAML, WS-Security, XML Security 등의 다양한 기술이 혼용된 형태로 적용되어 있다. 현재 그리드 보안 시스템에서 해결해야 하는 문제는 다음과 같다[8].

- 웹서비스 기반의 보안 프레임워크로 그리드 보안이 통합되면서 발생하는 문제를 해결하기 위해 개방형 환경을 위한 통합 보안 모델이 요구된다.
- 그리드 도메인의 특성상 다양한 협업이 필요한 만큼 사용자에 따라 다양한 접근권한이 적용되는 보안 메커니즘이 필요하다.
- 헬스케어 응용 도메인과 같이 민감한 접근제어가 필요한 응용 도메인에서는 GSI 보안 메커니즘이 유용하지 않으므로, GSI에서 사용되는 그리드 맵 파일의 문제를 해결할 수 있는 새로운 방안이 필요하다.

GSI는 그리드를 구성하고 있는 자원에 대한 접근 제어 메커니즘으로 그리드 맵 파일을 사용하여 사용자에게 권한을 부여하고 있다. 그리드 맵파일 방식은 기존의 유닉스 시스템에서의 계정 정책을 이용한 접근제어 방식이다. 웹서비스를 도입한 이후에는 SAML을 이용하여 별도의 정책 서버를 두고 접근 권한을 목록화한 뒤, SAML 프로토콜을 이용하여 그리드 서비스에 선언(assertion)하는 과정에 대해 정의하고 있다. 글로버스에서는 원격 사용자가 로컬 자원을 이용하기 위하여 각 자원의 실제 로컬 계정을 얻는 대신 그리드 맵 파일을 이용하여 사이트 계정을 획득한다. 그리드 사용자는 로컬 자원의 계정을 얻기 위해, 사용자의 DN(Distinguished Name)과 로컬 계정을 함께 기입한다. 그리드 맵 파일을 이용한 접근제어 매커니즘은 기본적으로 다수의 DN이 하나의 로컬 계정에 대해 결합할 수 있다. 이와 같은 방법을 사용함에 따라 다음과 같은 문제가 발생한다.

- 데이터 보안의 문제 : 여러 명의 그리드 사용자가 하나의 로컬 계정을 공유하게 되면, 특정 프로세스의 처리에 있어서 사용하고 있는 그리드 사용자를 구분할 수 없는 경우가 발생한다. 예를 들어, 헬스케어 시스템과 같이 데이터 사용이 민감한 응용의 경우, 임의의 사용자에 의한 데이터의 변형 또는 제거 등의 데이터 보안의 문제가 발생한다.
- 과금의 문제 : 그리드를 상용 서비스화 할 경우 과금 등의 문제에서는 특정 프로세스가 실제 어느 사용자에 의해 실행되는지를 구별해야 한다. 이 경우, 시스템 커널을 통하여 복잡한 과정을 거쳐 사용자를 알아내야 하며, 처리 불가능한 경우도 발생한다.
- 확장성의 문제 : VO간의 경계가 무너지는 웹서비스 환경에서는 기존의 GSI에서 사용하는 그리드 맵 파일로는 사용자를 관리할 수 없다. VO의 크기가 증가함에 따라 그룹을 관리하는 메커니즘이 필요하지만, 그리드 맵 파일로는 어떤 사용자가 어느 호스트로부터 인증을 받은 사용자인지를 구별할 수 없다. 한번 그리드 맵 파일에 등록된 사용자를 그룹별로 관리할 수 없기 때문이다.
- 상세한 접근권한 부재의 문제 : 그리드 맵 파일을 이용한 GSI 구조에는 정책을 적용하여 사용자별 접근권한을 설정하는 매커니즘이 존재하지 않는다. 매핑 과정 이후 로컬의 아이디가 가지는 권한을 원격지 사용자가 얻지만, 로컬 사용자별 접근권한에 관한 언급이 없는 상태이다. 즉, 로컬 정책을 적용하기 위해서는 기존의 GSI를 변경 또는 추가적인 기능을 마련해야 한다.

2.3 관련연구

[13]에서 제안한 FAS(Fedreation Agent Server) 모델은 CCM(Certificates Conversion Module), RDM(Role Decision Module), 그리고 ADM(Authorization Decision Module)으로 구성되었다. CCM은 프락시 인증서를 XML 형태로 변형하는 모델이며, RDM은 프락시 인증서를 기반으로 사용자의 역할을 결정하고 역할에 따른 로컬 정책을 결정하는 모듈이며, ADM은 로컬 자원을 요청하는 사용자의 접근권한을 결정하는 모듈이다. [13]에서 제안한 CCM은 프락시 인증서를 XML형태로 변형하여 로컬 관리자에게 편의성을 제공하고자 하였으나 구현과정에서 그리드 도메인 내에서 프락시 인증서 소유자만이 그리드 사용자로 인정되어 자원에 접근할 수 있는 한계 때문에 사용할 수 없었다. 또한 역할을 결정하는 RDM은 기존의 CCM이 프락시 인증서 변형 과정에서 RDM을 결정할 수 있는 역할을 추가하였으나 이를 제거함으로써 새로운 역할 결정 방안이 필요하였다. 기존의 RDM은 관리자가 CCM 과정에서 역할을 추가하고 결정한 역할에 해당하는 접근정책을 적용하는 과정이다. 그러나 이러한 방법은 관리자가 미리 사용자에 대한 정보를 모두 알고 있어야 한다는 점 때문에 사용자의 신분을 알아야하는 한계가 드러났다. 따라서 원격지 사용자의 그리드 인증서와 사용자의 역할을 결정지을 수 있는 새로운 방식의 RDM의 구현을 위해 본 논문에서는 신뢰 협상을 적용하였다. 이에 대한 상세한 내용은 3.3절에서 다루기로 한다.

3. 그리드 보안을 위한 신뢰 협상 모델

3.1 FAS(Federation Agent Server) 모델

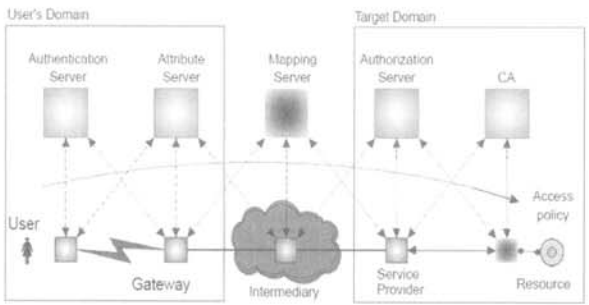
본 논문에서는 기존의 그리드 기반 보안 구조인 GSI의 단점을 극복하고 신뢰 협상 과정을 도입한 새로운 보안 메커니즘인 FAS 모델을 제안한다. FAS 모델은 권한 검증 메커니즘으로 RBAC(Role Based Access Control)을 확장한 접근 제어 메커니즘을 사용하여 사용자의 역할에 따라 상세한 접근권한을 결정하며, 신뢰 협상 과정을 통하여 사용자의 접근권한을 설정하는 통합보안 모델이다.

RBAC의 도입은 그리드 협업에서 중요한 관리의 편리함[9,10]을 제공한다. 어떤 조직 내에 임의의 역할이 생성되었을 때, 그 사용자가 접근할 수 있는 레벨이 결정되어야 한다. 이는 RBAC이 사용자와 허가 사이에 역할이라는 추가적인 수단을 적용하는데서 오는 이점이다. 어떤 역할에 할당된 허가는 조직의 보안 정책을 나타내는데, 예를 들어 누군가 조직에 새롭게 들어왔을 때나 조직 내에서 위치가 변경될 때, 관리자가 사용자에게 역할을 할당하는 일은 각 개인에게 직접 허가를 할당하는 것보다 더 쉽고 에러를 일으킬 확률도 낮다. 관리자적 측면에서 직원의 이동이 잦은 조직이거나 거대 시스템이라면 역할 기반 접근 제어는 특히 중요하다. RBAC에서 역할은 최소 특권의 원칙[11]을 사용함으로써 네트워크 내의 보안성을 향상시킨다. 최소 특권이란 한번 접근 요구사항이 결정되면 그 역할은 필요한 작업을 수행하기 위한 허가만을 얻어야 한다는 것을 의미한다. 추가적인 어떤 허가도 주어질 수 없다는 것이다. 역할 기반의 정책이 없는 네트워크에서 가끔 사용자들은 필요 이상의 접근허가를 가진다. 따라서 관리자들은 민감한 정보에 대하여 접근을 제한할 수 없는 경우가 발생한다. 그러나 RBAC을 도입함으로써 역할 밖의 정보에 대한 접근을 막을 수 있는 방안이 제공된다. 이러한 접근 거부는 보안 정책을 뚫으려는 악의를 가진 사용자를 막을 수 있다. 그리드는 다양한 응용에 적용되고 있고, 응용마다 서로 다른 목적으로 구현되므로 그에 따른 보안 메커니즘을 도입하는 것이 타당하다.

기존의 그리드 보안 구조에서, 그리드 자원에 접근하기 위한 권한은 UNIX OS의 파일접근 정책을 이용하여 구현되었다. VO(Virtual Organization) 그룹과 역할 정책은 그리드 자원에 대한 접근여부를 VO 사용자에 따라 서로 다르게 적용하지 않는다. 그리드 맵 파일을 이용한 방법에서는 VO 내의 모든 사용자들은 단일 사용자 계정으로 매핑 되었으며, 중복 매핑 또한 가능하였다. 이는 하나의 VO내의 모든 멤버들 사이에서 자원의 공유를 단순화하고, 개별적인 사용자 계정을 일일이 유지·보수해야 하는 관리차원의 오버헤드를 감소시키기 위해서였다. 결과적으로, 모든 접근은 접근권한의 전체 집합으로 용인되었으며, VO가 전체로서 접근하는 것으로 간주되어 그리드 사용자의 행동은 개별적으로 분리될 수 없다. 이와 같은 다-대-일 매핑은 상세한 접근 권한 정책을 구현하고 관리하기 위해서는 한계점을 가지고 있으며, 전체적으로 시스템 보안의 수준을 떨어뜨린다. 사용자나 어플리케이션, 자원 그리고 서비스가 증가할수록 이러한 낮은 보안의 레벨은 자원보안이나 데이터와 작업 보안 및 효율성 측면에서 위험요소를 안고 있다. FAS의 제안 배경은 다음과 같다.

① 그리드 맵 파일의 단점을 극복할 수 있는 접근권한 메커니즘의 필요성.
② 기존의 접근 권한 메커니즘으로는 민감한 자원을 보호할 수 있는 정책을 구현하기 어려움.
③ 역할에 따라 접근권한을 변경할 수 있는 상세한 접근(fine grained access)이 가능하면서도 VO를 관리할 수 있어야 함.
④ Grid 응용을 위한 접근권한과 정책을 모두 제공하는 통합 보안 모델의 필요성.

(그림 2)는 OGSA의 보안 드래프트[12]에서 사용자의 서비스 요청이 중개자를 통하여 전달되는 과정으로 가정하고, 사용자가 자신의 도메인의 인증 서버로부터 검증받은 인증서(credential)를 획득하고, 서비스 요청에 이를 이용하는 것을 시작으로 하는 시나리오이다. 사용자의 요청은 게이트웨이를 통하여 전달되고, 게이트웨이는 속성 서



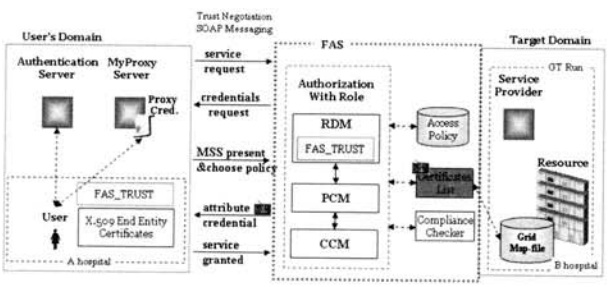
(그림 2) 중개자에 의한 서비스 요청의 전달

버에게 사용자의 특권 속성과 권리를 질의한 후, 요청을 포함한 선언을 중개자(intermediary)에게 전송한다. 이와 같이 전달된 요청은 중개자에게 전달되고, 타겟 도메인이 이해할 수 있는 형식의 선언으로 변환된다. 예를 들어, 사용자가 커버로스 티켓을 검증받은 인증서를 제시할 경우, 중개자는 타겟 도메인의 X.509 인증서의 형식으로 변환해야 한다. 또한 중개자는 요청이 전송될 때, 매핑 규칙이나 위임 정책을 포함하는 정책의 집합을 수용하고, 요청이 접수되었을 때, 타겟 도메인에서는 인증서를 유효화할 수 있게 된다. 성공적으로 유효성이 입증되면, 인증서를 기반으로 하는 아이디를 유도할 수 있으며, 로컬에서 정의한 접근권한에 관한 정책을 이용하여 요청한 사용자의 접근권한을 결정하게 된다.

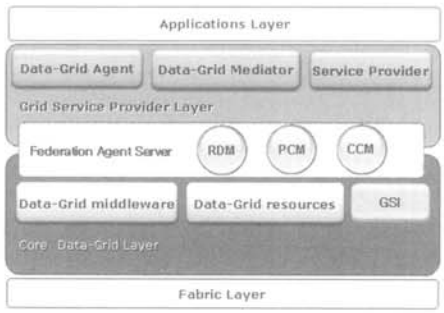
위의 예는 하나의 도메인 상에서 검증된 인증서가 중개자를 통하여 타겟 도메인에서 유효화될 수 있는 과정뿐만 아니라, 권한 위임의 문제를 비롯한 접근권한을 결정하는 전체 과정이 그리드 서비스가 제공되기 위하여 중요한 문제임을 나타낸다.

3.2 FAS의 구조

본 논문에서 제안하는 FAS 모델의 위치는 (그림 2)에서 중개자에 해당하는 계층에 위치한다. 위의 전체 과정을 통하여 FAS 모델에서는 사용자의 도메인에서 속성 서버에 대한 고려사항이 제외되었다. 제안하는 모델에서는 사용자의 특권 속성에 대한 내용은 정책파일을 노출하는 과정에서 사용자가 선택하도록 설계하였다. FAS모델의 전체구조는 (그림 3)과 같다.



(FAS_TRUST=>정의1, MSS(Minimal Satisfying Set=>정의2)
(그림 3) FAS 전체 구조도

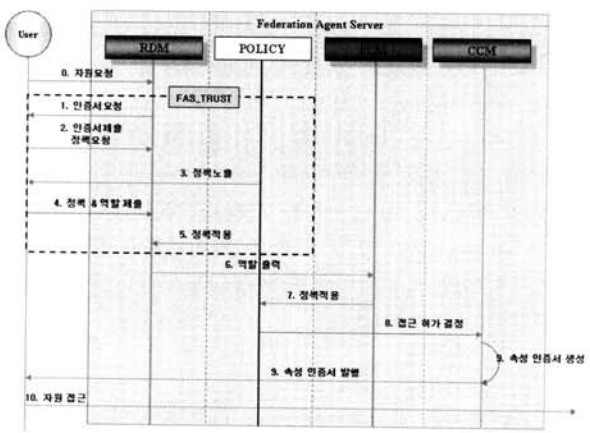


(그림 4) FAS의 계층적 구조

FAS 모델은 (그림 4)와 같이 그리드 미들웨어와 응용사이에 위치하여 독립된 계층으로 운영할 수 있도록 하였으며, 권한 설정을 위한 세 개의 모듈을 제안하여 사용자의 상세한 접근권한(fine grained rights)이 가능하도록 하고, 사이트 관리자가 동적으로 정책을 관리할 수 있도록 설계하였다.

3.3 FAS의 기능

본 논문에서 제안하는 FAS (Federation Agent Server) 모델의 3개 모듈, RDM, PCM, 그리고 CCM은 각각 다음과 같은 기능을 위하여 설계하였다. 우선 RDM에서 자원을 요청하는 클라이언트와의 협상을 통하여 클라이언트가 접근정책을 확인하고 자신의 역할을 선택하게 하였다. 이와 같은 방법은 기존의 모델에서 접근 권한을 서버로부터 일방적으로 할당받는 단점을 극복하고 사용자가 시스템에 유연하게 접근할 수 있는 메커니즘을 구현한 것이다. 둘째, PCM은 서버가 민감한 자원을 보호할 수 있는 방안으로 제안하였다. 즉, 역할에 따라 다양한 방법으로 접근허가를 제어할 수 있도록 하는 상세한 접근권한을 실현하였다. 마지막으로 CCM은 역할 속성을 추가한 속성 인증서를 클라이언트에게 발행함으로써 그리드 내에서 허가받은 접근권한을 기반으로 자원에 접근할 수 있는 메커니즘을 도입하였다. FAS 프레임워크의 동작 절차에 대한 전체 흐름도는 (그림 5)와 같다.



(그림 5) FAS 프레임워크 동작 절차

• RDM

RDM은 사용자가 제시하는 인증서를 이용하여 역할을 결정하는 모듈이다. 현재 FAS는 X.509 속성 인증서 [RFC3281] 표준을 따르도록 구현하였다. 제안하는 FAS 모델에서는 X.509 형식이 아닌 인증서(Credentials)에 대한 변환 기능은 제공하지 않는다. X.509 속성 인증서를 이용하여 신원기반의 프락시 인증서와 분리한 혼합 모드 접근권한 방식을 사용하였다.

클라이언트가 자원을 요청하게 되면, FAS는 정책 노출을 통하여 MSS(정의 2)를 요청하고, RDM 모듈을 통하여 역할을 결정하게 된다. RDM의 실행 과정은 다음과 같다.

(1) 서버(원격지 관리자)는 클라이언트(그리드 자원 요청자)에게 정책을 노출하고, 클라이언트는 MSS를 제시한다.

(2) 서버는 인증서를 확인한 후, 클라이언트가 제시한 인증서가 요청한 인증서가 맞으면 수용하고, 그렇지 않으면 거부한다.

(3) 요청한 인증서가 맞으면, 서버는 클라이언트에게 자원 접근권한에 관한 로컬 정책을 전송하고, 클라이언트가 선택한 역할을 입력으로 PCM 모듈을 수행한다.

(4) 요청한 인증서가 맞지 않으면, 서버는 클라이언트에게 접근 불가 메시지를 보낸다.

(5) 그리드 응용에 따라 필요한 또 다른 인증서가 있으면, (1)~(4)의 과정을 반복한다.

단, 본 논문에서는 서버와 클라이언트의 메시지 응답에 대한 실시간 보장, 오류 시 수정 등의 내용은 다루지 않으며, 본 논문의 연구 범위를 벗어나는 것으로 간주한다. 본 논문에서의 메시지 교환은 사용자의 인증서를 체크하여 서버가 원하는 인증서 인지를 결정하는 과정으로 그 범위를 한정하고 SOAP 통신을 이용하였다.

• PCM

PCM은 허가 조절 모듈로서 역할에 따른 각 자원에 관한 접근권한을 결정하는 모듈이다. PCM은 RDM을 통하여 출력으로 결정된 역할과 정책 파일을 입력으로 각 자원에 대한 허가를 결정한다. 역할 및 접근 허가에 관한 사항은 그리드 관리자가 직접 작성해야 하는 번거로운 부분이며, 각 응용에 따라 달라지는 사항이다. 관리자가 각 VO의 역할 추가 및 삭제, 그리고 갱신 등의 작업 및 자원의 접근허가에 관한 사항을 관리한다. 정책 파일은 XML을 이용하여 웹서비스가 가능하도록 하였다. PCM의 동작 과정은 다음과 같다.

1) 사용자의 역할을 확인한다.

2) 로컬 정책을 확인하고, 역할에 따르는 자원에 대한 허가를 매핑한다. 로컬 정책은 XML 파일로 구현되며, 자원에 대한 정책은 필요할 때마다 변경 적용한다.

3) 자원에 대한 역할의 매핑을 마치고, CCM으로 역할과 자원의 접근권한 XML 파일을 전송한다.

PCM을 통하여 각 자원에 대한 정책을 관리자가 생성, 갱신, 파기할 수 있어 역할 기반 모델의 관리 측면의 편리함과 더불어 정책 관리의 용이함이라는 장점을 제공하였다.

• CCM

CCM은 사용자의 역할에 따라 접근권한을 결정할 수 있도록 하는 속성 인증서를 발행하는 모듈이다. FAS 모델에서 사용하는 기본 인증서 타입은 X.509 V3로 최근까지 가장 잘 알려져 있고, 디지털 인증서의 포맷으로 널리 사용되고 있는 형식이다. 또한 그리드 인증서 포맷이기도 하다. X.509 최신 버전에서는 확장성을 제공하기 위하여 확장필드가 추가되어 필요한 요구사항을 구현할 수 있다. FAS의 CCM은 RDM을 통하여 결정된 사용자의 역할을 확인하고, PCM을 통하여 사용자에 해당하는 허가의 내용을 추가하여 속성 인증서를 발행한다.

3.4 FAS의 신뢰 협상

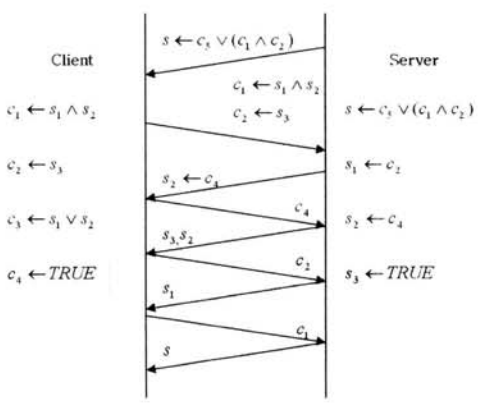
FAS 모델의 핵심은 협상과정을 통하여 클라이언트는 자신의 역할을 획득하고, 서버는 역할에 대한 자원의 접근권한을 할당할 수 있도록 한다는 것이다. RBAC 모델의 사용자와 역할 사이에 위치한 RDM은 FAS의 세 모듈 중에서 협상에 관여하는 모듈이다. 본 절에서는 "FAS_TRUST 프로토콜"을 제안한다.

일반적인 신뢰 협상 방법은 크게 다음과 같이 3 가지로 요약할 수 있다.

① 서비스를 요청하는 클라이언트가 자신의 모든 인증서를 노출하는 방법

② 인증서(또는 정책)의 민감성을 고려하여 노출해야 하는 인증서를 특화하는 방법

③ 상대방에 대하여 서로 간에 협상과 관련된 접근제어 정책을 노출하는 방법



(그림 6) 정책 교환 방식을 이용한 협상 전략

신뢰 협상 인증서를 교환하는 협상 전략을 정책을 교환하는 방법으로 표현하면 (그림 6)과 같다. 이와 같이 직접 인증서를 노출하는 방식이 아니라 상대의 정책을 확인하는 방식의 협상 전략은 인증서의 불필요한 노출을 막을 수 있다.

본 논문에서 제안하는 FAS 모델의 협상 프로토콜을 "FAS_TRUST"라 정의한다. 제안하는 FAS_TRUST 프로토콜이 고려하지 않는 사항들은 다음과 같다. 첫째, 신뢰 협상 과정에 대한 공격에 대해 보호할 방법은 FAS 모델이 제공해야 하는 보안 커뮤니케이션의 범주를 벗어난다. 둘째, 접근제어 정책이 변경되는 사항, 예를 들어 인증서에 대한 소유권, 로컬 컴퓨팅 환경에 대한 참조들, 정책에 두 가지 원칙이 함께 투입되는 경우 등에 대한 사항들은 고려하지 않는다. 셋째, 인증서를 획득하는 과정에 대해서는 논의하지 않으며 서버와 클라이언트의 메시지 응답에 대한 실시간 보장, 오류 시 수정 등의 내용은 다루지 않는다.

FAS_TRUST 프로토콜은 다음과 같이 정의한다.

협상을 시작하기 전에 클라이언트는 서비스 R에 접근하기를 원한다는 것을 표명하고, 원래의 자원 요청 메시지를 서버에 보낸다. 이 요청은 협상을 시작하게 한다. 즉, FAS_TRUST(POfRe, POfRo, R){}를 호출한다. FAS와 클라이언트 사이에 역할 선택에 관한 실패 메시지 또는 성공 메시지가 보내질 때까지 메시지를 교환한다. FAS_TRUST 프로토콜은 MSS를 확인한 이후 수행된다.

정의 2. (MSS : Minimal Satisfying Set).

최소 만족 집합은 하나의 정책을 만족하는 최소 인증서의 집합을 말한다. 즉, 부분집합이 없는 인증서의 집합으로, FAS의 최소 만족 집합의 필수 원소를 X.509 프락시 인증서, c_{proxy} 로 정의한다. 정리하면 다음과 같다.

MSS(minimal satisfying set) =>

$C^i \in C$ if and only if $g_{\Phi_s}(C^i)=1$ 을 만족하는 C^i의 부분집합 = $\{C^i, \varnothing\}$ 이고, $c_{proxy} \in C^i$ 이다.

정의 1. (FAS_TRUST PROTOCOL).

```
FAS_TRUST ( POfRe, POfRo, R ) {
입력 : POfRe : 로컬 자원에 대한 정책, POfRo : 역할에 대한 정책, R : Grid-PACS 도메인에서의 역할
출력 : 협상의 결과 ( FAIL or SUCCEED)
Let M, r
M : 노출 메시지 수열 초기값 (공집합);
r = NOT_TERMINATED;

If(R is a local resource) then
      r = FAS_send_response ( M, POfRe, POfRo, R );
         If (r == SUCCEED or r == FAIL) then // Negotiations 종료.
               return r;
 While (r == NOT_TERMINATED)
      Send message m to the other party;
      M= M+m ;
      r = FAS_check_for_termination( m, R); }
End of FAS_TRUST.
FAS_send_response ( M, POfRe, POfRo, R).
         Pm = ReceivedPolicy(M, POfRe, POfRo, R)
            // Pm = Role을 포함한 로컬 정책 파일 메시지, 1≤m≤n
         Choose any single message m' from Pm;
            M = M+m';
         r = FAS_check_for_termination(m',R);
            return r;
End of FAS_send_response.
FAS_check_for_termination ( m, R)
    If(R equals empty set) then return FAIL; //협상 실패
    If(R is element of m) then return SUCCEED; //협상 성공
         Return NOT_TERMINATED;
End of FAS_check_for_termination.
```

$c_{proxy} \in C'$ 이외의 C'의 원소는 각 사용자의 역할을 결정할 수 있는 디지털 인증서가 필요하다. 한 명의 사용자가 2개 이상의 역할을 결정할 수 있는 디지털 인증서를 소유할 수 있다. 예를 들어 Grid-PACS[14] 도메인이라 가정할 때, A에서 외과 전문의 인증서를 소유한 MD_AK (Kim as Medical Doctor in Grid Domain A)는 동시에 외과 병동의 과장임을 입증할 수 있는 인증서, HD_AK (Kim as Head of a department in Grid Domain A)를 소유할 수도 있다.

정리 1.

FAS_Trust 프로토콜을 만족하는 최소 만족 집합의 원소의 개수는 2 이상이다. 즉, MSS의 원소의 개수 ≥ 2.

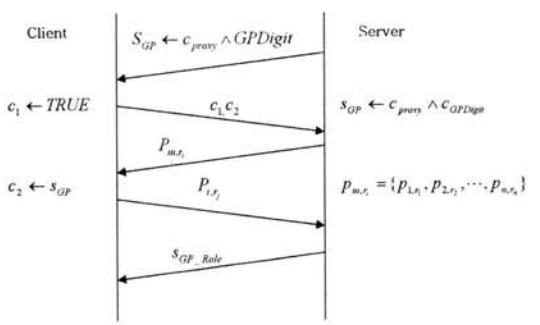
증명) 최소 만족 집합의 원소는 X.509 프락시 필수 인증서,c_{proxy} 와 하나 이상의 응용 도메인에서 통용되는 역할을 결정할 수 있는 디지털 인증서를 포함한다. 응용 도메인에서 통용되는 디지털 인증서를 $c_{GPDigit}$ 이라 할 때,

$\exists c_{proxy} \in C'$ 이고, $c_{GPDigit} \geq 1$ 이므로

$$\text{MSS}=\{c_{proxy}, c_{GPDigit_1}, c_{GPDigit_2}, \cdots, c_{GPDigit_n}\} \quad (4)$$

와 같이 나타낼 수 있고, 역할을 결정하기 위한 적어도 하나의 $c_{GPDigit}$가 필요하므로, 최소 만족 집합 MSS는 2 이상이다.

본 논문에서 제안하는 FAS 협상 전략은 (그림 7)과 같다. 단, s_{GP}는 응용 도메인에서 제공하는 서비스를 말하며, p_{m,r_i}는 FAS 내부 정책의 집합으로 $1 \leq m \leq n$, $1 \leq i \leq n$ 이다. m,i의 개수는 서버가 설계하는 정책과 역할에 따라 달라진다. c_1은 프락시 인증서로 클라이언트가 반드시 제시해야하는 디폴트 인증서이다. c_2는 병원에서 발급한 역할을 확인할 수 있는 디지털 인증서로 요청한 서비스를 획득하기 위한 MSS의 또 다른 원소이다



(그림 7) 제안하는 FAS 모델의 협상 전략

인증서 교환을 포함한 전체 FAS 협상단계는 다음과 같다.

[step1] 서버의 정책을 클라이언트에게 제시한다. 즉, Grid 서비스를 사용하기 위한 그리드 프락시 인증서, c_{proxy}와 역할을 할당할 수 있는 디지털 인증서, $c_{GPDigit}$를 요청한다.

[step2] 클라이언트는 자신의 로컬 디렉토리에서 인증서를 확인한 후 해당 인증서를 전송한다.

[step3] 서버의 역할을 포함한 정책 파일의 전체 집합을 전송한다.

[step4] 클라이언트는 자신의 역할을 포함한 정책 파일을 선택하여 서버에게 전송한다.

[step5] 서버는 컴플라이언스 체커를 통하여 클라이언트의 역할을 확인한다. 협상의 최종 단계에서 클라이언트는 자신의 역할을 할당받게 된다.

단 FAS 전체 과정 중, RDM에서의 결과를 즉시 사용자에게 전송하지는 않는다. PCM 및 CCM 과정을 거친 이후 속성 인증서를 발행함으로써 클라이언트는 해당 역할을 할당받게 된다.

그리드에서는 그리드 환경에서의 인증을 사용하므로 다수의 인증서를 원격지 서버에 노출하는 일은 필요하지 않다. FAS는 MSS를 확인, 즉 c_{proxy}와 $c_{GPDigit}$를 확인한 후, 자신의 정책 파일의 집합을 제시하고, 클라이언트는 자신이 접근하고자 하는 서비스(또는 자원)에 맞는 정책과 역할을 FAS로 전송한다. 이후 FAS는 컴플라이언스 체커를 통하여 사용자의 역할에 따른 자원 제공 여부를 결정하기 위하여 PCM을 적용한다.

클라이언트가 MSS를 제시하면, FAS는 MSS의 원소 c_{Proxy}와 $c_{GPDigit}$를 확인하는 절차를 거치게 된다. c_{Proxy}가 그리드 프락시 인증서가 맞는지를 체크해야하며, $c_{GPDigit}$를 통하여 클라이언트에게 할당할 수 있는 권한을 포함하고 있는 정책 파일을 전송해야 한다. 컴플라이언스 체커는 어떤 인증서가 정책을 만족하는지의 여부를 결정하여 Boolean 결과 값을 산출한다.

본 논문에서는 그리드 자원에 접근하기 위한 최소 인증서의 집합, MSS를 확인하고 자원을 노출할지의 여부를 결정하는 FAS_TRUST를 위한 새로운 타입의 컴플라이언스 체커를 제안한다. FAS를 위한 컴플라이언스 체커는 다음과 같다.

정의 3. FAS의 컴플라이언스 체커

FAS 컴플라이언스 체커는 함수 f_{GP}: $C \Rightarrow S$이다.
단, $C=\{c_{proxy}, c_{GPDigit}\}$이므로, 입력된 인증서의 집합이 S와 일치하는지의 여부만을 결정한다. 일치여부에 따라 f_{GP}의 결과는 참 또는 거짓이다.

일반적으로 정책을 교환하는 방식은 인증서를 교환하는 방식보다 더 복잡한 알고리즘으로 구성되는 것으로

알려져 있다. 본 논문에서는 Grid 도메인의 특성상 민감한 인증서(또는 정책)를 보호할 수 있는 방안으로 제안된 정책 교환 방식을 채택하였다. 그러나 각 응용에 따라 도메인이 한정될 수 있으므로, 필수 인증서 MSS를 정의함으로써, 협상 알고리즘의 단계를 감소시켜 복잡도는 줄이고, 불필요한 인증서의 노출을 방지하였다. 결론적으로, FAS_TRUST 협상 과정에서는 인증서의 최소 집합 MSS가 교환되는 인증서이며, FAS의 복수의 정책들이 한 번에 전송된다. 클라이언트는 정책 내의 역할과 자원에 관한 접근권한을 확인하고, $\forall p_{i,r_j} \in p_{m,r_i}$를 선택하면 FAS 컴플라이언스 체커를 거쳐 서비스를 제공할지의 여부를 결정하게 된다. 본 논문에서는 전적으로 실용화를 위하여 협상이 가능할 때마다 성공(완벽한 전략)하기 위한 목적과 협상의 복잡도를 줄이기 위한 방안으로 위와 같이 제안하였다.

4. FAS 모델의 분석 평가

4.1 FAS 모델의 검증

4.1.1 그리드 보안 측면의 분석평가

<표 1>과 <표 2>는 현재 글로버스에서 디폴트로 사용되고 있는 Pre-Web Services 기반의 보안 서비스와 FAS를 비교한 표이다. 웹서비스 이전 버전의 글로버스에서는 속성 인증서를 사용하지 않으며 응용 인터페이스를 지원하지 않는다. FAS 모델에서는 속성 인증서를 발행하여 관리자가 체계적으로 사용자를 관리하도록 하였으며, SOAP 메시지 보안을 비롯한 XML 기반 정책 기술을 사용함으로써 메시지 레벨 보안 기술을 적용하기 위한 웹서비스 기반의 보안 기술을 도입하였다.

<표 2>는 글로버스의 디폴트 인증 방식인 X.509 방식과 일반적인 유닉스 시스템에서 사용되는 ID/PW 방식 그리고 FAS의 권한검증 방식을 비교한 것이다.

기존의 글로버스 권한 검증 방식에서는 X.509 프락시 인증서만을 사용하였으나, 본 논문에서 제안하는 FAS에서는 속성 인증서를 추가로 발행하여 역할에 따라 자원에 대한 접근권한을 제한하였다. 권한 적용 방식에 있어서, 기존의 로컬 시스템 계정 정책에 따르는 획일적인 방식을 벗어날 수 있도록 협상을 통하여 역할에 따라 사용자가 서버의 자원 정책을 선택할 수 있도록 하였으며, 각 사이트 관리자가 권한을 관리할 수 있도록 하였다. 기존의 유닉스 계정의 파일 정책만을 적용하여 관리자 이외의 모든 클라이언트가 동일한 접근 권한 방식을 가졌으나, FAS 모델에서는 역할에 따라 자원 접근 권한을 가질 수 있도록 구현하였다.

FAS는 사이트 기반의 정책을 가지고 사용자 매핑이 가능한 시스템이다. 자원 관리자는 서로 다른 사용자 그룹에게 서로 다른 매핑 정책의 적용이 가능하다. 또한 필

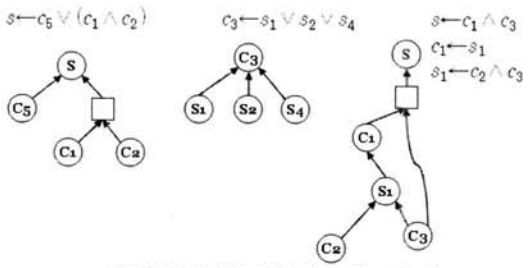
〈표 1〉 FAS 보안과 글로버스 보안 서비스의 비교

비교기준	Pre-Web Services 기반 보안	FAS 보안
보안 토큰	X.509 사용자 인증서 X.509 프락시 인증서	X.509 사용자 인증서 X.509 프락시 인증서 X.509 속성 인증서
속성 인증서	지원하지 않음	속성 인증서 발행
정책 적용	지원하지 않음	관리자를 위한 정책 편집 및 정책 에디터 지원
전송 보안	SSL/TLS (GT4.0)	WS-Security (SOAP 메시지 보안)
웹서비스 보안 기술	지원하지 않음	WS-Security, XML 기반기술
응용 인터페이스	지원하지 않음	• 사용자, 관리자 응용 서비스 모듈지원 • VO 접근제어 리스트 관리 및 UI 지원

〈표 2〉 FAS 권한검증과 기존의 글로버스 방식의 비교

비교기준	X.509 방식	ID/PW 방식	FAS 권한검증
권한검증 토큰	X.509 프락시 인증서	ID	X.509 프락시 인증서 X.509 속성 인증서
권한검증 방식	Subject DN과 ID 매핑	ID/PW	역할 기반의 권한 할당
권한 적용 방식	로컬 시스템의 계정 정책에 따른 방식	로컬 시스템의 계정 정책에 따른 방식	협상을 통한 역할에 따른 정책 파일 (사용자 선택방식)
권한 관리	지원하지 않음	지원하지 않음	사이트 관리자 기반의 권한 관리 지원
정책 관리	UNIX 계정의 파일정책을 이용 (관리의 개념 없음)	어려움	XML형식의 정책 파일을 적용한 다양한 정책관리
사용자 증가에 따른 관리	어려움	어려움	용이함

요하다면 새로운 사용자 그룹을 정의할 수 있다. 관리자는 또한 이러한 매핑이 사용될 호스트들의 그룹을 정의할 수 있다. 정책 매핑은 사용자의 역할이나 VO 내의 그룹을 포함하는 확장된 속성을 적용시킬 수 있다. 본 논문에서 제안한 FAS 모델은 역할에 따른 정책 파일을 사용자가 선택하게 함으로써 첫째, 협상의 과정을 단축하였고 둘째, 자원에 대한 다양한 접근제어 정책을 적용할 수 있도록 하였다.



(그림 8) 정책과 정책 그래프의 예

4.1.2 정책 그래프 전략 측면의 분석

(그림 8)은 정책과 대응하는 정책 그래프의 예이며, 정책 그래프는 두 가지 종류의 노드로 구성된다. 원형 노드는 인증서와 대응하는 노드이고, 네모 노드는 정책 내의 "∧" 연산자와 대응하는 노드이다.

(그림 9)는 정책 그래프를 구성하기 위한 Pseudo-code로 협상 그래프(N_Graph)의 입력으로 서버의 정책(P_s)과 클라이언트의 정책(P_c)을 사용한다. 그리고 모든 정책 그래프는 조건(condition)이 이분적인 일반적인 형태라고 가정한다.

정책 그래프를 구성하기 위한 알고리즘의 실행 시간은 P_s와 P_c의 정책의 개수와 길이에 달려있다. P_s의 정책의 길이를 l_{p_s}라 가정하면, $p_s \leftarrow c_5 \vee (c_1 \wedge c_2)$에서 p_s내의 총 인증서의 개수가 3개이므로 $l_{p_s}=3$이다. 모든 정책의 최대 길이를 L이라 하면, $L=\max_{p_s \in P_s \cup P_c} l_{p_s}$과 같이 나타낼 수 있다. (그림 9)에서 WHILE 루프, (4)-(11)에서 집합 E가 정책 내에 나타나는 인증서만을 포함하여 $|E| \leq (L+1)(|P_s|+|P_c|)$ 이므로, $(L+1)(|P_s|+|P_c|)$번 실행한다. 다음으로

```
N_Graph(P_s, P_c){
(1)   E={P_s를만족하는인증서}; // P_s: 요청한 서비스 s에 대한 정책
(2)   G=∅;
(3)   BuildGraph(G, target_P_s, condition_P_s);
(4)   While (E≠∅){
(5)     Pick e from E;
(6)     IF (p_e∈P_s∪P_c ){
(7)         BuildGraph(G, target_P_s, condition_P_s);
(8)         E=(E∪condition_P_s);}
(9)     ELSE{
(10)        BuildGraph(G, e, FALSE);
(11)        E=E ;
(12)    Return G; }
(13)
```

```
Build_Graph(G,target,condition) {
(14)    IF( target ∉ G ) Create a node for target;
(15)    IF( condition == FALSE){
(16)        prune(target, G);
(17)        return;}
(18)    FOR (each disjunct D of condition) {
(19)        IF (D is a conjunct consisting of more than one literals){
(20)        Create a rectangle node R in G;
(21)        Link R to the node of target;
(22)        For(each element e of D){
(23)            IF(e∈G) Link a direct edge from e to R;
(24)            ELSE Create a circle node for e
(25)                and link a direct edge from e to R; }}
(26)        ELSE{ // D 가 단일 인증서라면,
(27)            IF(D∈G) Link a direct edge from D to the node of target;
(28)            ELSE Create a circle node for D
(29)                and link a direct edge from D to the node of target;
(30) }}}
```

(그림 9) 정책 그래프 구성을 위한 Pseudo-code[14]

BuildGraph()과정에서 FOR 루프, (17)-(25)단계에서는 조건 내에서 하나의 분리(disjunction)를 처리하는 실행과정을 거치므로 적어도 L번 실행하게 된다. 정책 그래프가 $n \times n$ 인접 행렬로 표현가능하다고 가정하자. 여기서 n은 출력 그래프의 노드의 개수이다. 그래프, G(코드 번호 (13),(21),(24)에 해당)내의 임의의 노드는 $O(n)$의 복잡도를 요구한다. 2개의 인증서가 "∧"연산자를 나타내기 위해 필요하고, 길이 l을 갖는 정책은 $[l/2]$의 네모노드를 생성하기 때문이다. 이와 같이 결과 그래프는 $[L/2](|P_s|+|P_{cl}|)$ 개의 네모노드와 이와 대응하는 $(L+1)(|P_s|+|P_{cl}|)$ 개의 원형노드를 구성하므로

$$n \le [L/2](|P_s|+|P_{cl}|)+(L+1)(|P_s|+|P_{cl}|)$$

이고, 결과적으로 BuildGraph()의 실행 시간은 $L([L/2]+L+1)(|P_s|+|P_{cl}|)$이다.

따라서 (그림 9)에서 알고리즘의 총 실행 시간은

$$\text{total\_running\_Time} = L(L+1)([L/2]+L+1)(|P_s|+|P_{cl}|)^2 \quad (5)$$

이다.

본 논문의 FAS 모델에서는 MSS의 두 개의 원소에 해당하는 그리드 프락시 인증서와 역할을 구별할 수 있는 디지털 인증서를 사용하므로 협상과정에서 한 개의 네모 노드만이 존재한다. 따라서, 정책 그래프 기반의 전략 분석법을 이용한 FAS 협상의 총 실행 시간은

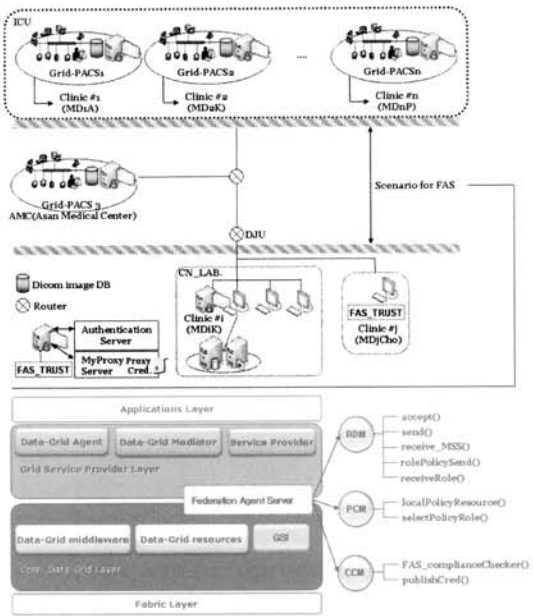
FAS_total_running_Time= $L[L/2](|P_s|+|P_{cl}|)$ 이므로

$$\frac{FAS\_total\_running\_time}{total\_running\_time} = \frac{L([L/2])(|P_s|+|P_{cl}|)}{L(L+1)([L/2]+L+1)(|P_s|+|P_{cl}|)^2}$$
$$= t\frac{1}{(|P_s|+|P_{cl}|)} \le 1 \quad (6)$$
$$(\text{단, } t = \frac{[L/2]}{(L+1)([L/2]+L+1)})$$

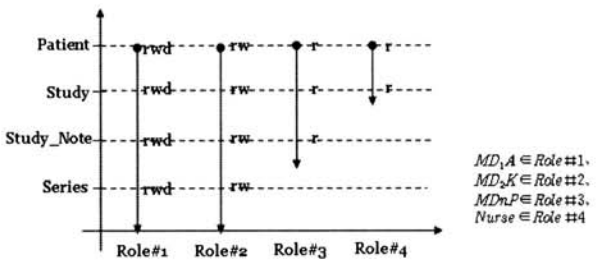
결론적으로, 일반적인 정책 그래프 기반의 총 실행시간과 비교하여 FAS 협상에서는 MSS를 사용함으로써 식 6과 같이 실행시간이 감소하는 결과를 얻었다.

4.1.3 FAS 적용 서비스 시나리오

FAS 모델 적용을 위한 Grid-PACS[15] 서비스 시나리오는 다음과 같다. Grid-PACS 노드의 구성은 (그림 10)과 같이 도메인 1 내에 n 개의 VO, 즉 Grid-PACS1 ~ Grid-PACSn 까지를 각각 하나의 병원으로 가정하고, 도메인2 내에 CN Lab.에서 의사 MDnK가 "Alice"의 X-ray 사진을 DICOM 이미지로 저장한 후, 각 병원 내에 MD1A(specialist), MD2K(general practitioner), MDnP(intern), 그리고 MDjCho(nurse)의 역할을 가진 사용자에게 FAS 모델을 통하여 서로 다른 접근 권한을 부여하여 Alice의 X-ray 이미지에 접근하는 시나리오로 가정한다. PACS (Picture Archiving and Communication System)에서 환자



(그림 10) PACS-Grid 전체 구성도



(그림 11) 각 역할에 해당하는 PACS 자원 접근정보

의 의료 영상 정보를 구분하는 기준에는 "Patient"와 "Study" 그리고 "Series"가 있다. "Patient"는 환자를 구분하는 ID로 사용되며 그 환자에 대한 의사의 전체적 소견을 관리하는 레벨로 구분된다. "Study"는 병증을 구분하는 코드 레벨을 통해 환자의 병증에 대한 정보를 확인 할 수 있다. "Study-Note"는 환자의 병증에 대한 소견을 작성하는 부분으로 Grid-PACS에서 추가된 내용이다. "Series"는 병증의 빈도에 대한 레벨 관리를 통해 환자의 병증의 정도를 확인 할 수 있다. 그리드 영역 안에서 의사들은 환자의 의료영상 정보와 함께 제공되는 "Patient", "Study", "Study-Note", 그리고 "Series"를 확인하여 다른 의사가 진단한 환자의 정보를 확인 할 수 있다. 환자의 정보에 관한 각 레벨간의 접근정책은 <표 3>, 각 역할에 따른 환자의 의료 정보에 대한 접근권한은 (그림 11)과 같다.

FAS 모델의 실험적 평가를 위하여 각 역할을 가진 서로 다른 도메인의 4명의 사용자, $MD_1A, MD_2K, MDnP, Nurse$를 대상으로 측정 I, II를 실행하였다. <표 4>의 측정 및 총 처리시간은 200회를 실행한 평균값이다.

- 측정 I : 원격지(ICU Grid-PACS1) 도메인의 FAS 클라이언트가 로컬 FAS(DJU. CN Lab)와 신뢰 협상을 통하여 역할을 할당받기까지 걸리는 시간 (RDM의 처리 시간)

〈표 3〉 Grid-PACS 서비스 시나리오의 역할에 따른 정책

Role #	각 역할에 대한 접근허가	비고
Role #1 specialist	<policy-Patient value="0xa" /> <policy-Study value="1x10xa" />... <policy-Study value="1x14xa" /> <policy-Study-Note value="1xa" /> <policy-Series value="2x0" />	Patient :0 Study:1 Series:2 disease #1:10 disease #2:11 disease #3:12 disease #4:13 disease #5:14 a:r-w-d b:r-w- c:r-- f:---
Role #2 generalpractitioner	<policy-Patient value="0xb" /> <policy-Study value="1x10xb" /> ... <policy-Study value="1x14xb" /> <policy-Study-Note value="1xb" /> <policy-Series value="2xb" />	
Role #3 intern	<policy-Patient value="0xc" /> <policy-Study value="1x10xc" /> ... <policy-Study value="1x14xc" /> <policy-Study-Note value="1xf" /> <policy-Series value="2xc" />	
Role #4 nurse	<policy-Patient value="0xc" /> <policy-Study value="1x10xc" />... <policy-Study value="1x14xc" /> <policy-Study-Note value="1xf" /> <policy-Series value="2xc" />	

〈표 4〉 RDM, PCM, CCM 동작 처리시간(단위:msec)

Role	클라이언트 위치	측정 I	측정 II	총처리시간
MD_1A	Grid-PACS$_1$ (Domain1-1)	6,588	826	7,414
MD_2K	Grid-PACS$_2$ (Domain1-2)	6,455	782	6876
MDnP	Grid-PACS$_3$ (Domain2-1)	138	295	433
Nurse	Grid-PACS$_4$ (Domain2-2)	194	421	489
평균 측정 시간		3,343.75	581.00	3,803.00

- 측정 II : 결정된 역할을 입력으로 자원에 대한 접근권한을 부여받고, 속성 인증서를 발행하는데 걸리는 시간(PCM, CCM의 처리 시간)

각 도메인 간 측정 I 처리의 평균 시간은 약 3,343msec, 측정 II 처리의 평균시간은 581msec이며, 평균 총 처리시간은 약 3,803msec으로 나타났다. FAS 모델의 각 모듈의 처리시간은 Grid-PACS를 상용화하더라도 사용자가 불편함을 느끼지 않을 정도의 처리시간임을 확인할 수 있다.

결론적으로 본 논문에서는 서로 다른 역할을 가진 사용자가 FAS 모델을 통하여 접근할 수 있는 자원을 제한하면서 Grid-PACS 도메인에 접근할 수 있는 메커니즘을 제공하였다. 그리고, FAS 모델의 CCM을 통하여 발급되는 속성 인증서에 사용기간을 명시함으로써 역할 및 접근권한을 할당받고 사용할 수 있는 시간을 조정가능하게 함으로써, 기존의 그리드 권한 검증 메커니즘으로 사용되는 CAS(Community Authorization Service)의 단점을 극복하였다.

또한 서로 다른 접근권한을 가진 사용자를 역할에 따라 구분함으로써, VO 내의 모든 사용자마다 상세화할 필요가 있는 정책 파일의 개수를 역할에 대한 그룹으로 분류하여 사용함으로써 사용자가 증가하더라도 관리가 용이하도록 확장성이 가능한 모델을 제시하였다.

5. 결론 및 향후 연구

초기의 그리드 기술은 과학적인 응용을 위한 컴퓨팅의 하부구조로 대학, 연구소 등 특정 커뮤니티에 의해 널리 이용되었으나, 최근 몇 년간 대형 벤더를 중심으로 그리드를 e-비지니스에 접목하려는 노력이 지속되었다. 그러나 과학적인 커뮤니티 외부의 사용자들에게 적용되는 그리드 기술은 매우 느리게 진행되고 있다. 그러한 주요 이유 중 하나는 보안과 신뢰의 부족을 들 수 있다. 그리드 기술이 상용화되기 위해서는 그리드의 다양한 협업 환경을 지원할 수 있는 보안 메커니즘이 구축되어야 한다. 그리드 보안에서 가장 중요하게 대두되는 이슈는 권한검증의 문제이다. 최근 그리드 연구자들은 "Grid-STP (Security, Trust and Privacy) 2007" 컨퍼런스를 발족하는 등 그리드 상에 보안 및 신뢰에 관한 분야를 새롭게 도출하고, 이러한 기술을 실현시키려하고 있으며, 여러 시스템들이 제안되었다. 제안된 각 시스템은 기존의 그리드 보안 구조인 GSI가 가지고 있는 문제점들을 해결하려고 노력하였으나, VO 관리나 정책 관리 등 편중된 문제해결을 위한 시스템들로 통합 솔루션을 제공하지 못하고 있다.

본 논문에서는 역할 기반 접근제어를 이용한 신뢰협상 모델, FAS(Federation Agent Server)를 제안하였다. 제안하는 FAS 모델은 기존의 RBAC(Role Based Access Control)

모델의 주요 요소인 사용자, 역할, 그리고 허가의 기본구조를 구체화하고 확장하여 연합 에이전트 서버를 설계함으로써 로컬정책에 따른 상세한 접근권한을 할당할 수 있는 시스템 독립적인 그리드 보안 통합 모델이다.

FAS는 각 사용자가 어떤 역할을 가지는지를 결정(RDM)하고, 역할에 따른 접근권한을 할당(PCM)하며, 역할과 접근권한을 포함하는 속성 인증서를 발행(CCM)하는 세 가지 내부 모듈로 구성된다. FAS는 사용자의 상세한 접근권한이 가능하도록 구현되었으며, 사이트 관리자가 동적으로 정책을 관리할 수 있도록 설계하였다.

첫째, RDM 과정에서 기존의 로컬 시스템 계정 정책에 따르는 획일적인 방식을 벗어날 수 있도록 협상을 통하여 역할에 따라 사용자가 서버의 자원 정책을 선택할 수 있도록 하였다. 또한 인증서 기반의 협상 방식이 아닌 정책 기반의 협상 방식을 채택하여 불필요한 인증서의 노출을 막고, 완벽한 전략을 위하여 최소 만족 집합, MSS(Minimal Satisfying Set)를 정의하였다. 최소 만족 집합을 통하여 일반적인 정책 그래프 기반의 신뢰 협상 방법보다 실행시간을 단축하였다.

둘째, 각 사이트 관리자가 권한을 관리할 수 있도록 PCM 과정에서 다양한 정책 적용을 가능하게 하였다. FAS의 PCM은 어떤 작업들을 어떤 사용자가 수행할 수 있으며, 어떤 우선순위를 가지는지에 대해 제한하기 위해서 서로 다른 사용자 그룹과 역할에 대한 다양한 정책을 적용한다. 이는 향상된 보안 레벨을 가지고 고품질 서비스를 제공하고 유지하기 위한 더 상세한 VO와 개별 사이트 모두에 관한 정책을 적용할 수 있는 장점을 가진다. PCM을 통하여 각 자원에 대한 정책을 관리자가 생성, 갱신, 파기할 수 있어 역할 기반 모델의 관리 측면의 편리함과 더불어 정책 관리의 용이함이라는 장점을 제공하였다.

향후, FAS의 기능을 추가하고, 기존의 사용자 프록시 인증서에 사용자 서명을 통하여 인증서 체인을 형성하는 방식의 그리드 "권한 위임(Delegation)" 문제를 웹서비스 기술로 대체하기 위하여 XML 형식의 정책파일을 통하여 해결하는 방안을 연구하여, 인증 및 권한검증뿐 아니라 권한 위임까지 FAS 모델에서 수용할 계획이다.

참 고 문 헌

[1] T. Ryutov et al., "Adaptive Trust Negotiation and Access Control for Grids," In Proc. of 6th IEEE/ACM International Workshop on Grid Computing, Seattle, WA, November, 2005.

[2] M. Winslett et al., "Negotiating Trust on the Web," IEEE Internet Computing Special Issue on Trust Management, pp.6(6):30 - 37, Nov./Dec., 2002.

[3] T. Yu et al., "Supporting structured credentials and sensitive policies through interoperable strategies in automated trust negotiation," ACM Transactions on Information and System Security, pp.6(1):1 - 42, Feb., 2003.

[4] M. Blaze, J. Feigenbaum and J. Lacy, "Decentralized trust management," In Proceedings of the IEEE Symposium on Research in Security and Privacy, (Oakland, CA), IEEE Computer Society, Technical Committee on Security and Privacy, IEEE Computer Society Press, pp.164-173, May, 1996.

[5] T. Yu, M. Winslett and K. Seamons. Interoperable strategies in automated trust negotiation. In Proc. of ACM Conference on Computer and Communications Security, Philadelphia, Pennsylvania, November, 2001.

[6] T. Yu, X. Ma and M. Winslett. "PRUNES: An efficient and complete strategy for trust negotiation over the Internet," In Proc. of ACM Conference on Computer and Communications Security, Athens, November, 2000.

[7] GGF Security Area, Grid Security Infrastructure Working Group, http://www.gridforum.org/security/gsi/index.html, 2007.

[8] Howard Chivers, "Grid Security: Problems and Potential Solutions," Department of Computer Science, University of York, 2003.

[9] D. F. Ferraiolo et. al, "A role-based access control model and reference implementation within a corporate intranet," ACM Transactions on Information and System Security, Vol.2, pp.34-64, Feb. 1999.

[10] R. Sandhu et. al, "The ARBAC97 model for role-based administration of roles," ACM Transactions on Information and System Security, Vol.2, pp.105-135, Feb., 1999.

[11] S. Osborn, "Mandatory access control and role-based access control revisited," In Proceedings of the 2nd ACM Workshop on Role-Based Access Control (RBAC-97), (New York, NY), ACM Press, Nov., 6-7, pp.31-40, 1997.

[12] Ian Foster et al., "Security Architecture for Open Grid Services," GGF OGSA Security Workgroup, June, 2003.

[13] H Cho, B Lee, K Lee, "A Trust Management Model for PACS-Grid," LNCS, Computational Science and Its Applications, ICCSA, 2007.

[14] Weifeng Chen et. al, "Optimizing Cost-sensitive Trust-negotiation Protocols," NSF Technical Report, 2004.

[15] Erberich SG et al., "Globus MEDICUS - Federation of DICOM Medical Imaging Devices into Healthcare Grids. Studies in Health Technology and Informatics," IOS Press, Vol.126, pp.269-278, 2007.

조 현 숙

e-mail : chojo@dju.ac.kr
1996년 대전대학교 수학과(학사)
2001년 대전대학교 정보통신공학과(석사)
2008년 대전대학교 정보통신공학과(박사)
2006년~현 재 대전대학교 교양학부 전임강사
관심분야 : 컴퓨터 네트워크 보안, 분산 컴퓨팅, 그리드 보안

이 봉 환

e-mail : blee@dju.ac.kr
1985년 서강대학교 전자공학과(학사)
1987년 연세대학교 전자공학과(석사)
1993년 Texas A&M 대학교 전기 및 컴퓨터공학과(박사)
1993년~1995년 한국통신 통신망연구소 연구원
1995년~현 재 대전대학교 정보통신공학과 교수
관심분야 : 그리드컴퓨팅, 네트워크보안, 웹서비스