논문 2010-47CI-6-3

클라우드 컴퓨팅 환경을 위한 상황인식 보안 시스템

(Context-Aware Security System for Cloud Computing Environment)

이 현 동*, 정 목 동**

(Hyundong Lee and Mokdong Chung)

요 약

클라우드 컴퓨팅 서비스 환경에서 인증 및 접근 제어와 같은 여러 보안 이슈가 발생하고 있다. 특히, 클라우드 컴퓨팅 환경에서 다양한 자원에 접속을 할 경우, 통합적으로 관리 및 제어가 가능한 인증 및 접근제어 모델이 필요하다. 이를 해결하기 위해서, 본 논문에서는 클라우드 컴퓨팅 환경에서 상황인식 기술과 통합인증 기술, 접근제어 기술, OSGi 서비스 플랫폼 기술을 접목하여, 상황인식 기반의 통합 인증(SSO) 및 접근제어 시스템을 제안한다. 또한 제안 시스템을 설계 및 구현함으로써, 클라우드 컴퓨팅 환경에서 상황에 따른 다양한 Multi Fact기반의 통합인증을 통하여 유연하고, 편리한 보안 시스템을 검증하였다. 이를 통하여 클라우드 컴퓨팅 환경에서 사용자 상황에 따라, 유연하고, 안전한 무중단 보안 서비스를 제공할 수 있음을 확인할 수 있었다.

Abstract

Many security issues occur in cloud computing service environment such as authentication, access control, and so on. In this paper, we propose an effective authentication and access control model which provide integrated management and control when we access various resources in cloud computing environment. To address these problems, we suggest a context-aware single sign-on and access control system using context-awareness, integrated authentication, access control, and OSGi service platform in cloud computing environment. And we show design and implementation of context-aware single sign-on and access control system. Also we verified the flexibility and convenience of the proposed system through multi fact based integrated authentication in cloud computing environment. We could provide flexible and secure seamless security service by user context in cloud computing environment.

Keywords : 클라우드 컴퓨팅, 상황인식(context-awareness), SSO, OSGi 서비스 플랫폼, 인증 및 접근제어

Ⅰ. 서 론

클라우드 컴퓨팅은 최근에 아마존, 마이크로소프트, 구글, 아이비엠 등 IT 관련 글로벌 기업들이 참여하면서 이슈화되기 시작하였다. 클라우드 컴퓨팅 환경은 애플리케이션을 개발하거나 서비스할 때 서버나 스토리지 등 컴퓨팅 자원 등을 자체적으로 보유하지 않고, 이 같은 자원을 갖고 있는 클라우드 컴퓨팅 플랫폼을 제공하는 회사의 자원을 이용해서 개발하고 서비스하는 것을 의미한다[1].

클라우드 컴퓨팅 서비스 환경에서는 가상화 기술 보안, 대용량 분산 처리 기술, 서비스 가용성, 대용량 트래픽 핸들링, 애플리케이션 보안, 접근 제어, 인증 및 암호와 같은 여러 보안 이슈가 발생하고 있다[2].

특히, 클라우드 컴퓨팅 환경에서 다양한 자원에 접속을 할 경우, 통합적으로 관리 및 제어가 가능한 인증 및 접근제어 모델이 필요하다.

이에 본 논문은 상황인식 기반의 통합 인증(SSO) 및 접근제어 시스템을 제안하고, 이를 설계 및 구현함으로

* 학생회원, ** 정회원, 부경대학교 컴퓨터공학과
(Dept of Computer Eng., Pukyong National University)

※ 본 논문은 중소기업청에서 지원하는 2009년도 산학연공동기술개발사업(No. 00038609-1)의 연구수행으로 인한 결과물임을 밝힙니다.
접수일자: 2010년9월23일, 수정완료일: 2010년10월25일

써, 클라우드 컴퓨팅 환경에서 유연하고, 편리한 보안 시스템을 검증하였다.

본 논문의 구성은 다음과 같다. Ⅱ장 관련 연구에서는 상황인식 보안 시스템에 사용되는 기본 기술들을 살펴보고, Ⅲ장에서는 상황인식 보안 시스템의 구조 및 각 컴포넌트 별 기능을 살펴본다. 그리고 Ⅳ장에서는 상황인식 보안 시스템의 구현 시나리오와 설계 및 구현 결과를 제시한다. 마지막으로 Ⅴ장에서는 결론 및 향후 연구 방향을 제시한다.

Ⅱ. 관 련 연 구

1. 상황인식 기술

상황인식(Context-Aware)기술은 변화하는 상황을 분석하여 사용자의 의도와 관련이 있는 정보인지를 판단하고, 유용한 정보이면 상황인식 응용을 실행하기 위해 정보를 요청하는 이벤트 신호를 발생시키는 기술이다[3]. 상황인식에 대한 개념은 1990년대 이래 분산 시스템에 의해 중요성이 증가되었으며 쉴 새 없이 변화하는 환경의 모바일 단말 사용에 수반되었던 수많은 문제점들에 대한 해결책으로 활용되고 있다. 상황 의존성은 클라우드 컴퓨팅 또는 유비쿼터스 컴퓨팅 시스템의 최근 연구 영역에서 주요한 이슈이다[4].

Schilit[5]는 최초로 상황 인식에 대한 용어와 정의를 소개하였다. Schilit는 네트워크 연결성, 통신 비용, 통신 대역폭 등 인근에 존재하는 자원인 컴퓨팅 상황정보(Computing Context Information), 사용자 프로필, 위치, 현재의 사회적 상황을 포함한 사람 정보인 사용자 상황정보(User Context Information), 조명, 소음수준, 교통 상황 등 물리적 상황정보(Physical Context Information)와 같이 세 가지 범주로 정의하였다. 또한 Chen과 Kotz[6]는 상황인식을 능동적(active) 상황 인식과 수동적(passive) 상황인식으로 구분하여 정의하였다.

2. 클라우드 컴퓨팅 환경에서의 보안

클라우드 컴퓨팅는 인터넷 기술을 활용하여 IT 자원을 서비스로 제공하는 것으로 소프트웨어, 스토리지, 서버, 네트워크 등의 자원을 필요한 만큼 빌려서 사용하고, 서비스 부하에 따라서 실시간 확장성을 지원받으며, 사용한 만큼의 비용을 지불하는 특징을 가진다[7].

클라우드 컴퓨팅 환경에서의 서비스는 대부분 웹 기반으로 제공되므로, 사용자의 신원을 확인하기 위한 인증이 반드시 필요하다. 현재 가장 활성화되어 있는 단순한 아이디/패스워드 기반의 인증 방식의 경우, 사용자들은 사용하는 서비스의 수만큼 아이디/패스워드를 생성 및 관리해야 함으로써 사용자들에게 많은 불편을 발생시킨다. 뿐만 아니라, 아이디 및 패스워드의 분실 및 도용으로 인해서 보안 취약성이 존재한다. 이를 해결하기 위해서 다양한 Multi Fact기반의 인증 방식의 적용이 필요하며, 가상화 자원에 대한 접근 제어 기술도 필요하다. 그림 1은 Gartner Report, UC Berkeley EECS, Cloud Security Alliance 자료를 바탕으로 클라우드 컴퓨팅 환경에서 핵심 보안 요소를 도출한 것이다[2].

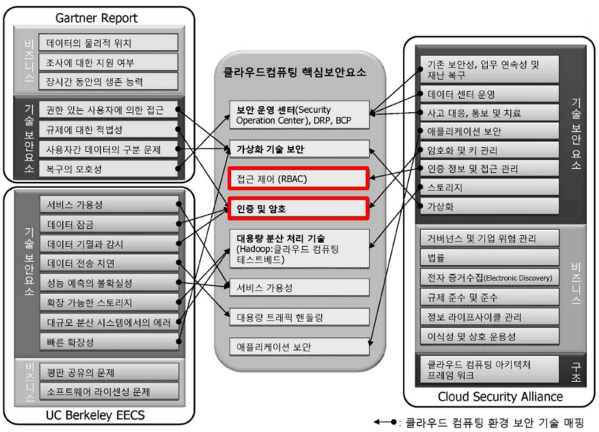

그림 1. 클라우드 컴퓨팅 환경에서의 보안 이슈
Fig. 1. Security issue in Cloud Computing Environment.

3. 접근제어 기술(RBAC, GRBAC)

RBAC(Role-based Access Control)는 사용자의 조직 상에서의 역할을 기반으로 접근권한을 특정 사용자가 아닌 해당 역할을 가진 사용자 그룹에게 부여하는 방식으로 조직의 구조와 연동하여 직책에 따라 보안 등급을 부여하며, 개별 사용자가 특정 직책을 부여 받으면 그에 상응하는 권한을 획득한다.

RBAC 기본 모델은 그림 2에서 보여주는 것처럼 컴퓨터 시스템을 통하여 시스템 내의 정보를 사용하는 객체로서의 사용자(U: user)와 시스템의 하나 또는 그 이상의 객체에 대한 특정 접근 모드(예 : read, write, update)의 승인을 나타내는 역할(R: role) 그리고 사용자 배정(UA: user assignment)과 인가 권한(P: permission), 세션(S: session)으로 구성된다[8].

다음 그림 2는 역할기반 접근 제어 모델을 나타낸다. GRBAC(Generalized Role-Based Access Control)는 상황에 근거한 접근제어를 수행하기 위하여, 접근 제어

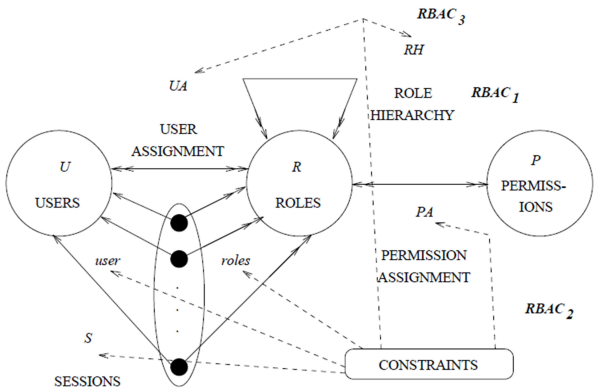

그림 2. 역할기반 접근제어 모델
Fig. 2. Model of Role-based Access Control.

결정에 사용자 역할(subject role), 객체 역할(object role), 환경 역할(environment role)을 추가하여 기존의 RBAC 모델을 확장하였다[9].

4. OSGi 서비스 플랫폼

OSGi(Open Service Gateway Initiatives)는 자바 기반으로 운영체제나 플랫폼에 독립적으로 운영되는 미들웨어 프레임워크로서, 네트워크 환경에서 서비스의 전달, 배치, 관리를 위한 표준 명세를 정의하는 기관인 OSGi 얼라이언스에서 개발한 개방형 표준이다[10].

기본적으로 OSGi 구조는 OSGi 컨테이너 구현체 위에서 컴포넌트(번들)들이 런타임 환경에서 동적으로 플러그인 되고 서비스 레지스트리를 통해 다른 번들에게 서비스를 제공하는 구조를 가지고 있다. 그림 3은 OSGi 서비스 플랫폼 구조를 나타낸다.

OSGi 서비스 플랫폼의 장점은 다음과 같다.

①자바 가상 머신(JVM) 기반의 플랫폼 독립적

②동적 서비스 변경과 같은 서비스 관리 기능 제공

③다양한 레벨의 시스템 보안 제공

④단일 게이트웨이 플랫폼에서 서로 다른 공급자로부터 제공된 여러 서비스 호스팅 기능을 사용 가능

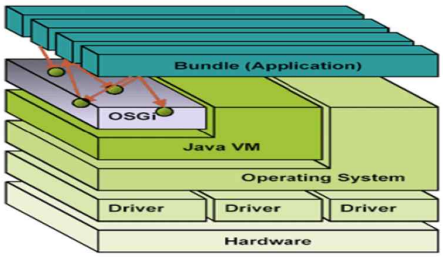

그림 3. OSGi 서비스 플랫폼
Fig. 3. Structure of the OSGi.

이러한 장점으로 인해서, OSGi 서비스 플랫폼은 제조업체, 서비스 공급자 및 개발자에게 네트워크로 연결된 다양한 장치들 간의 기본 프레임워크로 각광받고 있다[11].

Ⅲ. 상황인식 보안 시스템

1. 전체 시스템 구조

본 논문에서 제안하는 클라우드 컴퓨팅 환경을 위한 상황인식 보안 시스템은 OSGi 서비스 플랫폼 기반으로 SSO Manager, Context-Aware Manager, UI Manager 계층으로 구성한다. 다음 그림 4는 전체 시스템 구성을 나타낸다[12].

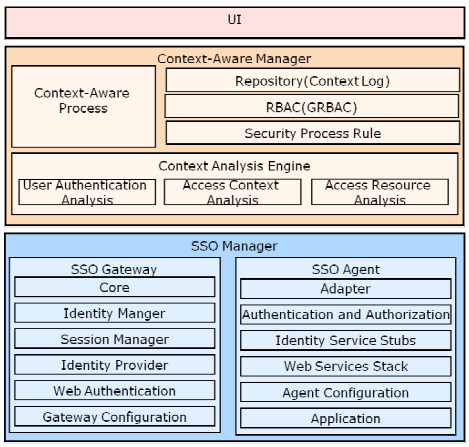

그림 4. 상황인식 보안 시스템 구성
Fig. 4. Context-Aware Security system Architecture.

가. SSO Manager

클라우드 컴퓨팅 환경에서 단일 인증(SSO)을 수행하며, SSO Agent와 SSO Gateway로 구성한다.

- SSO Agent: 사용자 프로그램이 실행되는 클라이언트 PC에 설치하며, 로그인시 사용자 인증 정보를 수집한다.

컴포넌트	기능
Adapter	각각의 애플리케이션을 동일한 표준으로 바꿈
Authentication and Authorization	사용자의 정보를 전달해 인증과 허가 요청
Identity Services Stubs	모든 애플리케이션에 사용할 수 있는 사용자 인증 정보
Web Services Stack	사용자가 사용하는 애플리케이션의 집합
Security APIs	보안을 위해 사용하는 라이브러리 모음
Application	서비스 제공자가 사용자에게 제공하는 App

- SSO Gateway: 인증 서버에 탑재된 인증 모듈로, SSO Agent로부터 사용자 인증 정보를 전달 받아

서 사용자 인증을 처리한다.

컴포넌트	기능
Core	인증 스킴과 Identity Manager와 Identity Provider를 통해 사용자 인증
Identity Manager	각각의 애플리케이션에서 사용되는 사용자 신원 관리
Session Manager	사용자가 인증 없이 각각의 애플리케이션을 이용할 수 있도록 세션 관리
Identity Provider	사용자 신원을 확인할 수 있는 정보를 코어에 제공
Web Authentication	웹을 통해 사용자 인증 과정을 연동
Gateway Configuration	관리자가 게이트웨이의 환경 설정을 관리

나. Context-Aware Manager

사용자 인증 정보를 상황에 따라 다양한 분석을 통하여, 인증 절차를 정의하는 기능을 수행한다.

- Context-Aware Process: 각종 상황정보의 분석 결과를 취합하고, 분석된 결과에 따라 보안 처리 절차를 Security Manager로 전달한다.
- Repository: 사용자 인증 시 획득된 상황정보의 로그를 저장하는 저장소 역할을 하며, 자원에 대한 사용자 선호도를 선정하는데 유용한 자료로 활용한다.
- RBAC(GRBAC): 사용자 권한 별로 자원의 접근 제어 가능 여부를 정의한다.
- Security Process Rule: 인증 및 접근제어 처리 절차를 정의한다.
- Context Analysis Engine: 신원적 상황, 물리적 상황, 역사적 상황, 정서적 상황, 자원적 상황을 분석한다. 또한, OSGi 서비스 플랫폼으로 구성되어 있어, 각각의 상황 분석 모듈을 자유롭게 추가, 수정, 삭제가 실시간으로 가능하다.

컴포넌트	기능
User Authentication Analysis	사용자 권한을 분석
Access Context Analysis	사용자가 인증시의 물리적 상황(단말기 성능, 배터리상태, 사용자위치, 네트워크 보안상태), 역사적 상황(접근 시간), 정서적 상황(사용자 선호도) 정보를 분석
Access Resource Analysis	접근하려는 자원에 대한 접근 권한을 분석

다. UI Manager

UI Manager는 사용자 인증 웹페이지를 제공하는 기능을 수행하며, 기본적으로는 아이디/패스워드 기반의 인증 UI를 제공하며, 추가적으로 OTP 인증 UI 및 클라우드 컴퓨팅 자원의 접근 거부 페이지를 제공한다.

2. 시스템 내부 데이터 흐름

그림 5는 상황인식 보안 시스템의 상황정보 처리를 위한 내부 데이터 흐름을 나타낸다.

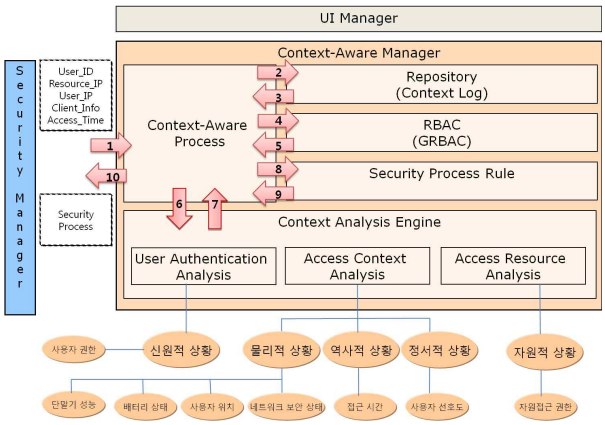

그림 5. 내부 데이터 흐름도
Fig. 5. Internal data flow.

상황인식 보안 시스템의 내부 데이터 흐름을 살펴보면, SSO Manager에서 사용자 로그인 시 획득한 정보인 사용자 아이디, 사용자가 로그인하는 클라이언트 PC의 IP 와 자원정보, 접근하려는 자원의 IP 정보와 접근 시간 정보를 Context-Aware Manager로 전달한다.(1번 과정)

Context-Aware Manager에서는 획득한 다양한 상황정보를 바탕으로 아이디/패스워드 기반의 기본 인증 이나, 추가 인증(아이디/패스워드 + OTP) 또는 접근 차단의 적용 여부를 분석한 후(2번 과정-9번 과정), SSO Manager로 전달한다.(10번 과정)

3. 상황 정보 분류

상황인식 보안 시스템의 상황 정보는 신원적 상황,

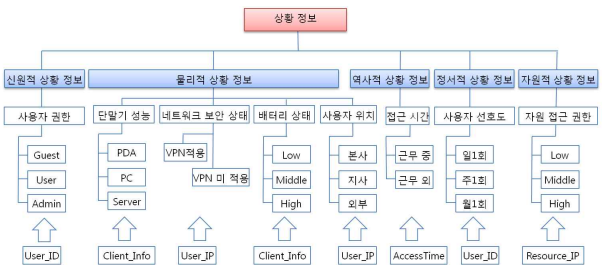

그림 6. 상황정보 분류
Fig. 6. Classification of the context information.

물리적 상황, 역사적 상황, 정서적 상황, 자원적 상황으로 분류한다. 그림 6은 상황정보 분류를 나타낸다.

4. 통합인증 및 접근제어 알고리즘

다음 표 1은 상황인식 보안 시스템의 인증 및 접근제어 처리 절차를 나타낸다.

표 1. 인증 및 접근제어 처리 절차
Table 1. Authentication and access control process.

Notations	
ID_u	사용자 ID
PW_u	사용자 패스워드
$RESIP_u$	접근하려는 자원의 IP
$UserIP_u$	사용자 IP
$ClientInfo_u$	사용자 사용 단말기의 상태 정보
$Time_u$	접근 시간
SP_u	보안 처리 절차 (ID, OTP, Deny)
Detailed Protocol	
(1) UI → SSO Manager: ID_u//PW_u//ContextInfo($RESIP_u$,$UserIP_u$,$ClientInfo_u$,$Time_u$) (2) SSO Manager: Check[ID_u//PW_u] (3) SSO Manager → Context-Aware Manager: ID_u//ResultofCheck[ID_u//PW_u]//ContextInfo($RESIP_u$,$UserIP_u$,$ClientInfo_u$,$Time_u$) (4) Context-Aware Manager: Analysis[ID_u//ContextInfo($RESIP_u$,$UserIP_u$,$ClientInfo_u$,$Time_u$)] (5) Context-Aware Manager → SSO Manager: SP_u (6) SSO Manager → UI: SP_u (7) UI: UI[SP_u]	

MAUT(Multi-Attribute Utility Theory)는 다중변수에 대한 의사결정 문제(decision problem)에서 유틸리티(utility)를 통한 정략적인 의사결정방법이다. 의사 결정자가 원하는 제비뽑기(lottery)의 결과에 대한 개인의 선호도(preference)를 유틸리티 수로 나타내고, 유틸리티 함수(u(x))는 가치 x에 대해 주관적인 가치(utility)를 나타낸다.

유틸리티는 0과 1사이의 상대적인 값으로써 $u(x^0)$, $u(x^*)$를 각각 가장 낮은 인증 절차 유틸리티와 가장 높은 인증 절차 유틸리티라고 두면 $u(x^0)=0$, $u(x^*)=1$로 나타낸다[13~14]. 예를 들면, 신원적 상황, 물리적 상황, 역사적 상황, 정서적 상황, 자원적 상황을 속성(Attribute)으로 평가될 때 보안 처리 절차 결정을 위한 전체 유틸리티 함수는 다음 식 1과 같이 정의된다.

$$u(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} k_i u_i(x_i), \sum_{i=1}^{n} k_i = 1 \quad (1)$$

(k_i : 가중치, $u_i(x_i)$: 속성x_i에 대한 유틸리티 함수)

표 2. 상황인식 알고리즘
Table 2. Context-aware algorithm.

```
MAUT(X)
for i = 1 to n
 if Repository == null then
  ask the user's preference and decide k_i;
 else update k_i; // reference of repository(UserID, Location,
AccessTime)
 u(x_1,x_2,…,x_n) = k_1u_1(x_1)+k_2u_2(x_2)+… +k_nu_n(x_n)
   // k_i: set of positive scaling constants for all i
   // x_i: domain dependent variable, where u_i(x^o_i)=0,u_i(x^*_i)=1
 do u_i(x_i) = GetUtilFunction(x_i) end
 if u(x_1,x_2,…,x_n) == u(x_sp) then
   return sp;
end;
```

```
GetUtilFunction (x_i)
 // Determine utility function due to users' preferences
 // x_i is one of domain dependent variables
 uRiskProne : user is risk prone for x_i // convex
 uRiskNeutral : user is risk neutral for x_i // linear
 uRiskAverse : user is risk averse for x_i //concave
 x: arbitrary chosen from x_i
 h: arbitrary chosen amount
 <x+h, x-h> : lottery from x+h to x-h
 // where the lottery (x^*, p, x^o) yields a p chance at x^*
 // and a (1-p) chance at xo
 ask user to prefer <x+h, x-h> or x // interaction
  if user prefer <x+h, x-h> then
    return uRiskProne;        // e.g. u = b(2^cx-1)
  else if user prefer x then
    return uRiskAverse;       // e.g. u = blog_2(x+1)
  else
    return uRiskNeutral;
end; // e.g. u = b
```

상황인식 보안 시스템은 표 2와 같이 사용자가 처한 각종 환경적인 데이터를 상황 정보로 인식하고, RBAC(GRBAC)를 통해 정량적 값으로 변경한다. 이 값을 MAUT 알고리즘으로 전달하고, 이 과정에서 각 속성에 가중치를 곱해서, 사전에 정의한 보안 처리 절차 값과 비교하면서, 다양한 상황의 조건 값들 중에 사용자의 선호도에 가장 근접한 보안 처리 절차 값을 결정한다.

MAUT 알고리즘에서는 유틸리티 함수 u가 모험 회피(risk averse), 모험 중립(risk neutral), 모험 노출(risk prone) 중에서 어떤 것인지 결정한다[14]. 우선 임의의 x와 h에 대해서 의사 결정자의 선호도를 확인하는데, 제비뽑기 <x+h, x-h>와 기대결과(expected consequence) x 중에서 어떤 것을 좋아하는지 확인한다. 제비뽑기 <x+h, x-h>는 같은 확률로서 x+h와 x-h를 선택할 수 있다는 것을 의미한다. 이런 시험을 여러 다른 x와 h 값에 대해서 반복한 결과, 만약 의사 결정자가 주로 제

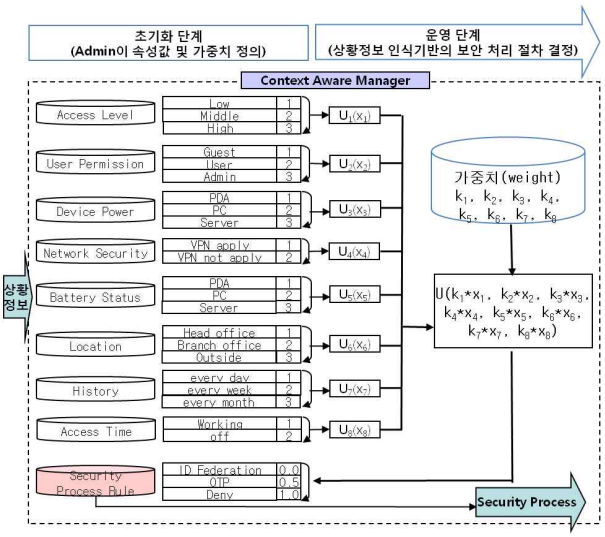

그림 7. MAUT를 통한 보안 등급 결정 프로세스
Fig. 7. Security level decision process using MAUT.

비뽑기를 선호하면 모험 노출이라고 추정할 수 있고, 기대 결과 x를 선호하면 모험 회피라고 간주 할 수 있다. 제비뽑기와 기대 결과 사이에 특별한 선호가 없으면 모험 중립이라고 본다.

그림 7은 MAUT 알고리즘을 통한 보안 등급 결정 프로세스를 나타낸다.

초기화 단계에서 관리자(Admin)는 Context-Aware Manager의 “Security Process Rule” 모듈을 통해서 보안 처리 속성 값(아이디/패스워드 이나 OTP기반의 추가 인증 또는 접근거부)을 정의하고, “Context Analysis Engine” 모듈에서 상황 정보를 처리하기 위한 상황 정보의 가중치(weight)를 정의한다.

운영 단계에서는 “Context Analysis Engine” 모듈에 있는 상황 정보 처리 번들을 통해서 사용자가 처한 환경적인 데이터를 상황정보로 인식하고, RBAC(GRBAC) 모듈을 통해 정량적 값으로 변경한다. 이 값을 MAUT 알고리즘으로 전달하고, 이 과정에서 각 속성에 가중치를 곱해서, 사전에 정의한 보안 처리 절차 속성 값과 비교하면서, 다양한 상황의 조건 값들 중에 사용자의 선호도에 가장 근접한 보안 처리 절차를 결정한다. 또한, 이 과정에서 RBAC (GRBAC) 모듈에서 Access Level 및 User Permission 조회를 통해서 자원 접근에 대한 권한 분석이 이루어진다.

사용자 인증 시 획득한 상황 정보(User ID, Location, Access Time 등)는 Repository 모듈에 저장하며, 이는 Context-Aware Manager가 자율적으로 속성 값 범위

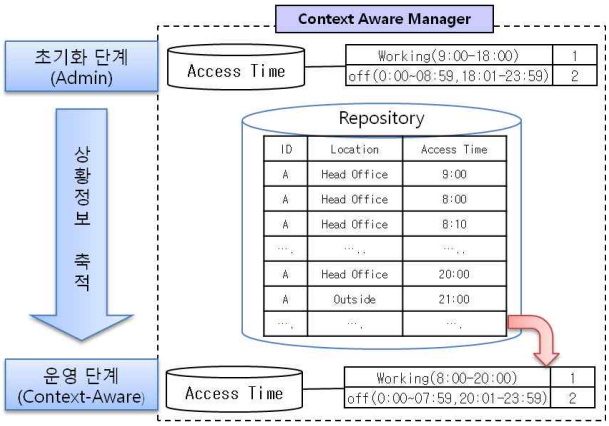

그림 8. 상황정보 속성 값의 자율적인 변경
Fig. 8. Adaptive change of context-information attribute values.

및 가중치를 변경하는데 활용한다.

그림 8은 상황정보 속성 값의 자율적인 변경 개념을 나타낸다.

예를 들어, ① 초기화 단계에서 관리자가 Access Time의 근무 시간 속성 값의 범위를 9:00-18:00이라고 정의하고, 상황인식 보안 시스템을 운영한다.

② 운영 단계에서 많은 상황 정보가 Repository 모듈에 축적된다. 사용자 A의 경우 축적된 위치정보, 접근 시간 정보를 분석한 결과 평균적인 근무 시간이 8:00-20:00이었다고 가정하자.

③ Context-Aware Manage에서는 사용자 A의 Access Time의 근무 시간속성 값의 범위를 8:00-20:00로 자율적으로 변경한다.

Ⅳ. 상황인식 보안 시스템 설계 및 구현, 평가

1. 구현 시나리오

클라우드 컴퓨팅 환경을 위한 상황인식 보안 시스템의 구현 시나리오는 그림 9와 같다.

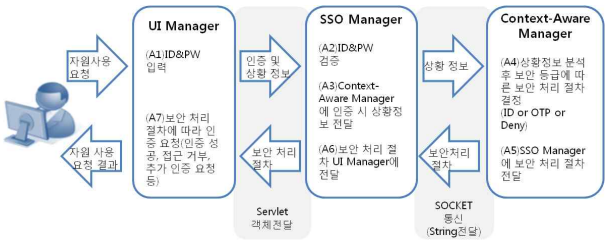

그림 9. 구현 시나리오
Fig. 9. Implementation of a scenario.

①클라우드 컴퓨팅 환경에서 사용자는 아이디/패스워드 기반으로 상황인식 보안 시스템에게 사용자 인증을 요청한다.

②상황인식 보안 시스템은 Context-Aware Manager에 의해서 신원적 상황 정보(사용자 권한), 물리적 상황 정보(단말기 성능, 배터리 상태, 사용자 위치, 네트워크 보안 상태), 역사적 상황(접근 시간), 정서적 상황(사용자 선호도), 자원적 상황(자원별 접근 권한)을 종합적으로 판단하여, “아이디/패스워드 기반으로 인증을 할 것인지?”, 아니면, “추가적인 인증 수단[OTP(One Time Password), PKI(Public Key Infrastructure), 경량 PKI 등]을 요청할 것인지?” 또는 “접근 거부를 할 것인지?”를 결정한다.

③상황인식 보안 시스템에서 추가적인 인증 수단을 사용자에게 요청할 경우, 사용자는 요청받은 인증 수단을 제시하고, 인증 성공 시 사용자는 클라우드 컴퓨팅 자원을 사용할 수 있다.

④만일, 추가적으로 다른 자원을 사용할 경우에는 자원의 접근 권한을 실시간으로 파악한 후, 추가 인증을 요청 한다.

2. 상황인식 보안 시스템 설계 및 구현

OSGi 서비스 플랫폼으로 Context-Aware Manager로 개발하였기 때문에, 동적으로 상황정보 처리 모듈을 변경할 수 있다. 그림 10은 상황정보 처리 모듈의 동적 변경 구성을 나타낸다.

SSO Gateway는 SSO Agent로 부터 사용자 인증 시 획득한 상황정보를 Context-Aware Manger의 Context-Aware Process에 전달하고, Context Analysis Engine는 신원적 상황정보, 물리적 상황정보, 역사적 상황정보, 정서적 상황정보, 자원적 상황정보를 분석한다.

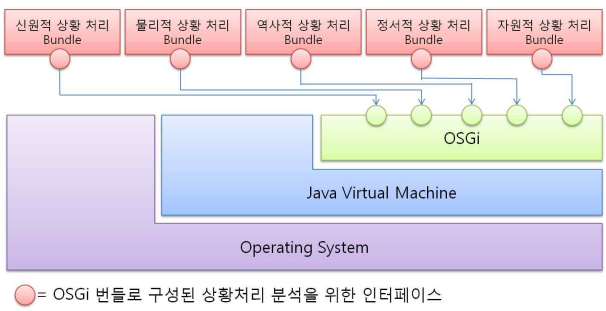

그림 10. 상황정보 처리 모듈의 동적 구성
Fig. 10. Dynamic configuration of context-information process module.

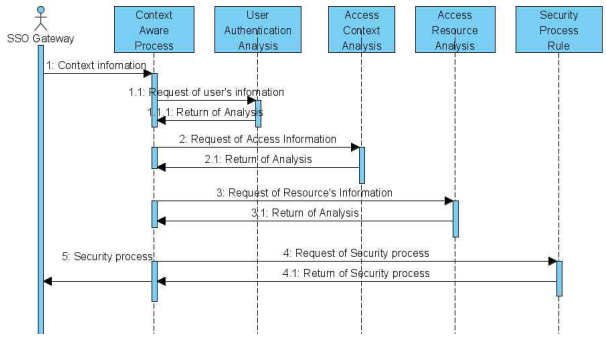

그림 11. 시퀀스 다이어그램
Fig. 11. Sequence Diagram.

그림 11은 상황인식 보안 시스템의 시퀀스 다이어그램(Sequence Diagram)이다.

상황정보 분석 완료 후, 사전에 정의되어 있는 보안 처리 절차에 따라 인증 성공 또는 추가 인증 요청 또는 접근 거부를 클라우드 컴퓨팅 자원 사용자에게 전달한다.

Context-Aware Manager의 구성 정보는 별도의 XML 파일로 작성하여, 각 상황정보 요소별 가중치 수정을 원활하게 하였으며, RBAC기반의 DB를 제어하기 위해서 iBatis 프레임워크를 활용하였다. 이를 통하여, 데이터베이스에 접근할 때 필요한 자바 코드를 현저하게 줄일 수 있고, XML를 사용하여 간단하게 SQL Statement에 매핑 시킴으로써, SQL 쿼리 튜닝을 쉽게

표 3. 상황인식 보안 시스템 실행 결과
Table 3. Typical result of Context-Aware Security System.

[Case1] • User’s ID: “user1” [User Permissions: user. History: every day] • Resource’s IP: “203.250.123,180” [Access Level: Low] • Client IP: “202.250.123,100” [Network: VPN_O, Location: Head office] • Client Info: “PC”, “High” [Device: PC, Battery Status: High] • Access Time: ”09:00:00” [Access Time: working] **※ Analysis result: 0.0 [Security Process: ID Federation]**
[Case2] • User’s ID: “user2” [User Permissions: admin, History: every day] • Resource’s IP: ”203.250.123,190” [Access Level: High] • Client IP: “202.30.34.2” [Network:VPN_X, Location: Outside] • Client Info: “PC”, “High” [Device: PC, Battery Status: High] • Access Time: “09:00:00” [Access Time: working] **※ Analysis result: 0.75 [Security Process: OTP]**
[Case3] • User’s ID: “user2” [User Permissions: admin, History: every day] • Resource’s IP: ”203.250.123,190” [Access Level: High] • Client IP: “202.30.34.2”[Network:VPN_X, Location: Outside] • Client Info: “PC”, “High” [Device: PC, Battery Status: High] • Access Time: “07:00:00” [Access Time: off] **※ Analysis result: 1.00 [Security Process: Deny]**

할 수 있다.

클라우드 컴퓨팅 환경을 위한 상황인식 보안 시스템의 실행결과는 표 3과 같다.

사용자 인증 시 획득한 상황정보인 User's ID, Resource's IP, Client IP, Client Info, Access Time에 따라서 보안 처리 절차[“아이디/패스워드”, “추가 인증(OTP)”, “접근 거부(Deny)”]를 결정한다. Case2와 Case3을 비교해보면, Access Time이 다르다는 것을 확인 할 수 있다. 제안하는 상황인식 보안 시스템은 같은 사용자가 동일한 상황이라고 할지라도, Access Time에 (“근무 시간 중”, “근무 시간 외”) 따라서 보안 처리 절차를 다르게 결정한다.

3. 제안 시스템 평가 및 논문의 기여도

소프트웨어 품질의 특성 및 척도에 대한 표준화인 ISO9126 Quality Model[15]의 6가지 품질 속성(Functionality, Reliability, Usability, Efficiency, Maintainability, Portability)을 기반으로 본 논문에서 제안하는 상황인식 기반 통합 인증 시스템은 기존의 단일 인증 시스템을 비교하면, 표 4와 같다.

본 논문에서 제안하는 클라우드 컴퓨팅 환경을 위한 상황인식 보안 시스템이 기여한 부분은 다음과 같다.

①클라우드 컴퓨팅 환경에서 상황정보 분류 및 MAUT 알고리즘을 사용한 상황인식 기술과 Multi Fact 기반의 통합인증 기술, RBAC와 GRBAC의 접근제어 기술, OSGi 서비스 플랫폼 기술을 통합 인증 및 접근제어 시스템에 접목

표 4. 기존 단일 인증 시스템과 제안 시스템의 비교
Table 4. Comparison between a single authentication system and the proposed system.

특징	단일 인증 시스템	제안하는 상황인식 보안 시스템
기능성	단일 인증	Multi Fact기반의 다중 인증
신뢰성	패치 및 결함 수정 시 시스템 중단	OSGi 플랫폼 도입으로 무중단 통합인증 및 접근제어 서비스 제공
사용성	고정적인 인증 서비스 제공	다양하고 유연한 인증 및 접근제어 서비스 제공
효율성	다양한 상황이 발생하더라도 동일한 인증 제공	자원 및 사용자 등급에 따른 인증 절차의 다양화로 인증 시 발생하는 비용 최소화
유지 보수성	패치 및 결함 수정 시 시스템 중단	OSGi 플랫폼 도입으로 무중단 통합인증 및 접근제어 서비스 제공
이식성	단일 플랫폼에서 동작	OSGi 플랫폼 기반으로 임베디드 시스템 등 다양한 플랫폼에서 동작 가능

②다양한 상황 정보(신원적 상황정보, 물리적 상황정보, 역사적 상황정보, 정서적 상황정보, 자원적 상황정보 등)를 분석하여 보안 처리 절차를 통합인증 및 접근제어 시스템 내에서 자율적으로 결정 가능

③클라우드 컴퓨팅 환경에서 사용자 상황에 따라, 유연하고, 안전한 무중단 보안 서비스를 제공할 수 있는 보안 시스템 모델 제시

V. 결론 및 향후 연구

클라우드 컴퓨팅 환경에서 자원의 접근 권한 및 사용자 인증은 단일 시스템에 비해서 유연하고, 자동화된 통합 인증이 필요하다.

이에 클라우드 컴퓨팅 환경에서 상황인식 기술(MAUT, 상황정보 분류)과 접근 제어 기술(RABC, GRBAC), 통합 인증 기술(아이디/패스워드, OTP, PKI, SPKI 등), 런타임 환경에서 동적으로 플러그인 되는 OSGi 서비스 플랫폼 기술을 접목하여, 유연하고 자동화된 통합 인증 및 접근제어 시스템을 설계 및 구현하였다.

본 논문에서 제안한 클라우드 컴퓨팅 환경에서 상황인식 보안 시스템의 장점은 다음과 같다.

①다양한 Multi Fact기반의 통합 인증 절차를 지원(아이디/패스워드 기반 + OTP + PKI, + SPKI 또는 접근 거부)함으로써 클라우드 컴퓨팅 환경에서의 강력한 통합 인증 및 접근제어 서비스를 제공한다.

②무중단 시스템을 적용하여, 운영 중이라도 동적으로 상황처리 Rule 정의 및 인증 및 접근제어 모듈을 자유롭게 추가 삭제 가능(OSGi 서비스 플랫폼)하다.

③상황 인식 기술을 통하여, 사용자의 상황에 따른 다양한 인증 방법, 사용자 편의성 및 시스템 안전성을 보장한다.

향후 연구계획은 상황인식 기반의 계산 알고리즘 보완과 다양한 접근 모델 및 보안 정책 수립을 중점적으로 연구 개발하고자 한다.

참 고 문 헌

[1] 김진택, “클라우드 컴퓨팅 기술 및 표준화 동향”, *TTA Journal No.125*, 42-47쪽, 2009년

[2] 김명주, “클라우드 컴퓨팅 보안 요소”, *제1회 클라우드 컴퓨팅 정보보호 기술 워크샵*, 140-154쪽, 2009년

[3] 백종훈, 윤병주, “모바일 디바이스에서 상황인식

컴퓨팅을 위한 사용자 활동 상태 추정", *전자공학회 논문지*. 제43권 SP편, 제1호, 67-74쪽, 2006년 1월
[4] 류승완, 장효선, 신동천, 박세권, "상황인식 컴퓨팅 기술 동향", *주간기술동향*, 통권, 1435호, 2-3쪽, 2010년
[5] Schilit, B., Adams, N. Want, R., "Context-Aware Computing Applications", *Proc. of the 1st International Workshop on mobile Computing Systems and Applications*, pp.85-90, 1994.
[6] Guanling Chen and David Kotz, "A Survey of Context-Aware Mobile Computing Research", Dartmouth Computing Science, 2000.
[7] James Staten, "Is Cloud Computing Ready for the Enterprise", Forrester Research. http://www.forrester.com/rb/research, 2008.
[8] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, Charles E. Youman, "Role-Based Access Control Models," *Journal of IEEE Computer*, Vol. 29, No. 2, pp. 38-47. 1996.
[9] M.J.Covington, et al., "Generalized Role-Based Access Control for Securing Future Applications", *Proc. of 23rd National Information System Security Conference(NISSC)*, Baltimore, pp. 115-125, Oct. 2000.
[10] OSGi Alliance, "OSGi Service Platform Core Specification", Release4, Version4.1, April. 2007.
[11] 김석구, "유니버셜 미들웨어 OSGi 최신 기술 동향", *지능형 홈네트워크*, 6호, 24-37쪽, 2007년
[12] 이현동, 정목동, "클라우드 컴퓨팅 환경에서 상황인식 기반 통합인증 시스템 설계 및 구현", *한국지능정보시스템 학회 2010년 춘계학술발표대회 논문집*, 163-169쪽, 2010년 6월
[13] 양석환, 정목동, "이기종 네트워크에서 퍼지 알고리즘과 MAUT에 기반을 둔 적응적 보안 관리 모델", *전자공학회논문지*, 제47권 CI편, 제1호, 104-115쪽, 2010년 1월
[14] R. L. Keeney and H. Raiffa, *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, Cambridge university press, pp. 261-271, 1993.
[15] ISO 9126 Software Quality Characteristics, http://www.sqa.net/iso9126.html

저 자 소 개

이 현 동(학생회원)
2001년 경성대학교 컴퓨터공학과 학사 졸업.
2007년 부경대학교 컴퓨터공학과 석사 졸업.
2009년 9월～현재 부경대학교 컴퓨터공학과 박사 과정.
<주관심분야 : 보안, 상황인식, RFID/USN/RTLS>

정 목 동(정회원)-교신저자
1981년 경북대학교 컴퓨터공학과 학사 졸업.
1983년 서울대학교 컴퓨터공학과 석사 졸업.
1990년 서울대학교 컴퓨터공학과 박사 졸업.
1984년～1985년 금성반도체(주) 연구소 연구원
1985년～1996년 부산외국어대학교 컴퓨터공학과 교수
1999년～2000년 아이오와 주립대학교 방문교수
1996년～현재 부경대학교 컴퓨터공학과 교수
<주관심분야 : 컴퓨터응용보안, 인공지능, 상황인식 컴퓨팅>