공 학 석 사 학 위 논 문

# FCM 알고리즘과 퍼지 결정트리를 이용한 RFID/USN 환경에서의 상황인식 보안 서비스


2009년 2월

부 경 대 학 교 대 학 원

정보보호학 협동과정

양 석 환

공 학 석 사 학 위 논 문

# FCM 알고리즘과 퍼지 결정트리를 이용한 RFID/USN 환경에서의 상황인식 보안 서비스


지도교수 정 목 동

이 논문을 공학석사 학위논문으로 제출함.

2009년 2월

부 경 대 학 교 대 학 원

정보보호학 협동과정

양 석 환

# 양석환의 공학석사 학위논문을 인준함

2009년 2월 23일


주 심 공학박사 신 봉 기 인

위 원 공학박사 송 하 주 인

위 원 공학박사 정 목 동 인

# 목 차

# 그 림 목 차

# 표 목 차

# Context-Aware Security Service in RFID/USN Environments using FCM Algorithm and Fuzzy Decision Tree

**Seok Hwan Yang**

*Department of Interdisciplinary Program of Information Security, Graduate School,*

*Pukyong National University*

Abstract

Ubiquitous technology has penetrated into almost every aspect of modern life, spreading to the furthest reaches of the world. And the research of the security technology to solve the weakness of security of the ubiquitous environment is paid to attention. The ubiquitous environment based system needs a security service which might be changed and adapted by various user context. A lot of system, however, have a security service using fixed rules in real world, thus many systems can not solve the more complex situation. Therefore we try to research context aware security service.

Many existing researchs on context aware security service are using

ACL (Access Control List) or are based on RBAC (Role Based Access Control). But they have a overhead in the management of security policy and also have problem, that is, they can not handle unexpected situation in the real world. This thesis proposes a context-aware security service providing multiple authentications and authorization from a security level which is decided dynamically in a context-aware environment using FCM (Fuzzy C-Means) clustering algorithm and Fuzzy Decision Tree algorithm consequently.

This thesis provides experimental data for proposed model. The experimental data shows suggested model could solve typical conflict problems of RBAC system due to fixed rules and could improve overhead problem on the management.

Suggested model draws security level about current situation using the distance between the center of cluster and the various context information, and then determines final security level using Fuzzy Decision Tree. And the suggested model supports diverse security services by using security level. We expects the suggested model to be applied to the diverse applications using contexts of user such as healthcare system, emergency system, and so on.

# Ⅰ. 서론

최근 무선 기술 및 다양한 센서 기술의 발달에 힘입어 인간 중심의 컴퓨팅 환경이라고 할 수 있는 유비쿼터스 기술이 점점 보편화되고 있다. 이런 유비쿼터스 기술의 발달로 인해 언제 어디서나 모든 정보를 공유할 수 있고 누구나 쉽게 접근할 수 있는 유비쿼터스 환경이 점차 구축되고 있으며, U-시티, 스마트홈, 헬스케어 시스템과 같은 다양한 센서가 제공하는 정보를 이용한 시스템의 개발이 확산되고 있다. 그러나 언제 어디서나 원하는 정보를 얻을 수 있는 유비쿼터스 환경의 특징은 개인정보 유출 및 다양한 보안문제를 발생시켰고, 이러한 보안 문제의 해결을 위하여 유비쿼터스 환경에서의 보안기술 연구가 점차 주목받고 있다.

유비쿼터스 환경 구축의 기반을 이루는 요소 중의 하나인 센서 네트워크는 온도, 습도, 조도, 영상 및 사용자의 이동상황, 건강상태와 같은 정보를 시스템에 제공하는 다양한 센서들로 구성되어 있다. 그러나 이러한 센서들이 시스템에 제공하는 사용자의 주위환경 및 상황에 대한 정보는 사용자의 프라이버시 침해 문제를 일으킬 가능성이 크다. 따라서 이러한 정보의 접근에 대하여 강력한 보안기능을 제공할 필요가 있다.

그러나 모든 정보가 사용자 프라이버시 보호에 중요한 것은 아니다. 또한 모든 상황에서 강력한 보안기술을 적용하는 것은 오히려 시스템의 효율성 저하와 함께 사용자의 불편을 가중시킬 수 있다. 특히 사용자의 건강을 관리하고 생명을 보호하기 위한 헬스케어 시스템에서는 긴급 상황 시, 강력한 보안보다는 사용자의 생명을 보호하는 일을 우선적으로 고려하여야 하는 것과 같다.

이러한 문제를 해결하기 위하여 다양한 상황에 따라 유연한 보안 서비스를 제공할 수 있는 상황인식 보안 서비스의 필요성이 부각되고 있다.

## 1. 상황인식 보안 서비스

상황인식 보안 서비스 (context-aware security service)란 상황 인식 컴퓨팅 (context-aware computing)을 통하여 다양한 상황에 대해 가장 적절한 보안 서비스를 유연하게 제공하는 서비스를 말한다. 상황 인식 컴퓨팅이란 유비쿼터스 환경에서 임의의 애플리케이션이 사용자의 다양한 환경 요소에 대한 상황 정보 (contextual information)를 감지하여, 사용자가 이를 이용할 수 있도록 해 주는 컴퓨팅 패러다임이다. 상황인식 컴퓨팅은 1994년 Schilit와 Theimer에 의하여 최초로 논의되었으며, Schilit와 Theimer는 상황 (context)이란 사용 장소나 주변의 사람, 사물의 집합 또는 시간이 지나면서 발생하는 변화 등을 일컫는다고 정의하였다[1,2]. 또한 Dey는 상황이란 엔티티의 상태를 특징지을 수 있는 모든 정보를 말하며, 여기서 엔티티란 사람, 또는 사용자와 애플리케이션 간의 의사소통에 관계되는 사물을 말한다고 정의하고 있다[3].

상황인식 컴퓨팅은 현실에서의 다양한 상황을 정보화하고 이를 활용하여 사용자 중심의 지능화된 서비스를 제공하는 컴퓨팅 방식으로, 넓은 의미에서 볼 때 유비쿼터스 컴퓨팅의 일부분으로 볼 수 있다. 그러나 접근 방법적인 면에서 볼 때 상황인식 컴퓨팅은 실세계의 상황을 표현하는 정보기술에서 시작한다는 것이 유비쿼터스 컴퓨팅과 다르다. 그러나 유비쿼터스 환경의 구축을 위해서는 상황인식 컴퓨팅 기술이 반드시 필요하므로 상황인

식 컴퓨팅 기술은 유비쿼터스 컴퓨팅 기술의 핵심기술의 하나라고 볼 수 있다[1].

유비쿼터스 환경은 음악, 영화와 같은 다양한 멀티미디어 컨텐츠, PDA, 휴대폰 등 다양한 유무선 기반의 자원들이 공존하므로, 사람과 컴퓨팅 기기 및 환경이 서로 상호작용을 통하여 상황정보를 생성 및 통합하고, 통합된 정보를 바탕으로 하는 적응적 보안 서비스가 요구된다. 이러한 보안 서비스를 상황인식 보안 서비스라고 한다.

## 2. 기존의 상황인식 보안 서비스의 문제점

상황인식에 관한 연구는 마크 와이저의 유비쿼터스 컴퓨팅 개념이 제시된 이후로 많은 연구가 진행되어 왔다. 그러나 상황인식 기술을 컴퓨팅 보안에 적응하는 상황인식 보안 연구는 비교적 최근에 시작되었고 이로 인해 많은 상황인식 보안 연구는 서로 유사한 방식을 응용하여 진행되어온 경향이 있다. 상황인식 기술을 보안에 적용한 대표적인 예로서 Illinois 대학에서 수행하고 있는 Gaia[4] 프로젝트의 Cerberus와 Georgia Tech에서 제안한 CASA (Context-Aware Security Architecture)[5] 등을 들 수 있다. 표 1은 Cerberus와 CASA의 정성적인 비교 내용을 보여준다[6].

[표 4] Cerberus와 CASA의 정성적 비교

| 분류 | 기준 | Cerberus[4] | CASA[5] |
|---|---|---|---|
| 사용자 인증 | 인증 신뢰도 관리 | 가능 | 가능 |
| | 인증 신뢰 산출 공식 | 없음 | 없음 |
| | 보안 등급 속성 | 미적용 | 미적용 |
| 접근 제어 | 접근 제어 모델 | ACL | GRBAC |
| | 권한 할당 | 사용자 | 역할 |
| | 데이터 추상화 | 미제공 | 제공(사용자, 객체, 환경정보) |
| | 조직 구조 반영 | 미반영 | 가능 |
| | 정책 구성 오버헤드 | 적음 | 도메인 수에 따라 증가 |
| | 정보의 기밀성 유지 | 불가 | 불가 |

Cerberus는 ACL (Access Control List)을 기반으로 접근제어를 수행한다. ACL은 전통적인 접근제어 정책 모델로서 간단한 정책의 구성이 가능하지만 데이터 추상화 기능을 제공하지 않기 때문에 보안 서비스를 제공하고자 하는 조직의 권한 및 책임 구조를 보안 정책에 적용하기 어렵다.

CASA는 GRBAC (Generalized Role-Based Access Control)을 기반으로 접근제어를 수행하므로 사용자, 객체, 환경정보들을 역할로 추상화하여 보안 정책 관리기능을 제공한다. 그러나 GRBAC는 RBAC가 제공하는 권한 추상화를 제공하지 않으며, 관리 도메인 요소들이 많아짐에 따라서 관리해야할 역할의 양이 많아진다. 또한 객체 역할과 환경 역할 간의 중복되는 영역에 의해 발생되는 충돌 문제로 인해 보안정책 관리 및 구성에 따르는 오버헤드가 많이 발생하게 되며, 보안정책의 설정 여부에 따라 예상하지 못한 상황, 즉 보안 정책에 등록되지 않은 상황에 대한 대응이 어렵다는 단점을 가진다.

현재 수행되고 있는 상황인식 보안연구는 Cerberus 및 CASA와 같이

ACL, GRBAC 등을 적용하고 있는 경우가 많으며, Cerberus 및 CASA와 동일한 문제에 노출되어 있다. 따라서 이러한 문제점을 해결할 수 있는 상황인식 보안 서비스에 대한 연구가 요구된다.

## 3. 연구의 목적, 방법 및 논문의 구성

본 논문에서는 기존의 상황인식 보안 연구가 가지는 시스템 관리에 대한 오버헤드 발생 문제 및 고정된 규칙의 적용에 따른 한계를 해결하기 위하여 센서의 정보를 이용하여 사용자의 상황을 인식하고, 사용자가 처한 상황에 따라 다양한 수준의 보안기술을 유연하게 적용할 수 있는 USN/RFID 환경에서의 상황인식 보안 서비스를 제안한다.

제안 모델은 FCM 클러스터링 알고리즘을 이용하여 자율적인 분류 기능을 통해 도출하기 위한 보안 등급의 개수만큼 클러스터를 생성한다. 또한 퍼지 결정트리를 이용하여 다양한 상황정보에 대한 보안 등급을 통합하여 최종적인 보안 등급을 도출한다. 도출된 보안 등급에 따라 다양한 보안 서비스를 제공한다. 또한 각 도메인에 대하여 특화된 접근권한을 부여할 수 있도록, 도출된 보안 등급에 따른 접근권한 목록 관리 기능을 제공한다.

논문의 구성은 다음과 같다. 2장에서는 상황인식 보안 서비스모델에 대한 기존 연구 내용 및 관련 연구 내용과 배경 이론에 대하여 살펴보고, 3장에서는 FCM 클러스터링 알고리즘을 이용한 상황인식 보안 서비스 모델을 소개한다. 4장에서는 제안한 상황인식 보안 서비스 모델을 직접 구현하고 실험한 결과 분석을 다루고 마지막으로 5장에서는 결론 및 향후 연구 방향을 제시한다.

# Ⅱ. 관련 연구

이 장에서는 RFID/USN 환경에서의 상황인식 보안 서비스 모델에 대하여 기존에 수행되었던 연구내용과 관련된 배경이론을 살펴봄으로써 본 논문에서 제안하는 상황인식 보안 서비스 모델의 정당성을 살펴보고자 한다.

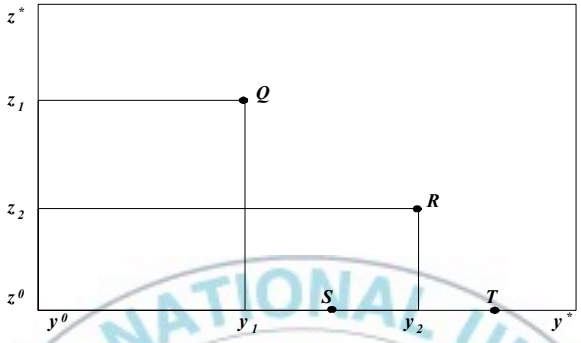
## 1. MAUT와 GRBAC를 이용한 상황인식 보안 시스템

### 1.1 MAUT (Multi Attributes Utility Theory)

MAUT[7,8,9,10,11]는 다양한 속성에 따른 의사 결정 방법의 하나로 의사 결정자는 결과에 대한 개인의 선호도를 0과 1사이의 유틸리티 수로 표현하며 결과 유틸리티의 최소값은 $u(x^0)=0$, 최대값은 $u(x^*)=1$로 나타낼 수 있다. 기대 유틸리티를 최대화 시켜주는 유틸리티 수를 결과에 대입하며, 의사 결정자의 최적 행동의 기준이 된다. 유틸리티 함수는 일반적으로 다음과 같은 과정을 통하여 선정된다.

**선정준비** : 결과 $Q$를 실수 $x$에 사상시키는 평가 함수를 $X$라고 하면 $x=X(Q)$ 이다. 의사 결정자가 결과 $x_1$과 $x_2$ 중에서 어떤 것을 선호하는지 확인해 볼 수 있다. 그림 1은 두 개의 특징 변수 $Y$와 $Z$에 대한 결과 공간을 보여주고 있는데, 결과 $T$와 결과 $S$ 중에서 의사 결정자가 선호하는 것

을 질의를 통해서 확인해 볼 수 있다. 결과 $Q$는 $y=y_1$이고 $z=z_1$인 결과이고, 결과 R은 $y=y_2$이고 $z=z_2$인 결과이다.




[그림 1] 두 특징 변수 결과 공간

**독립성 확인 :** $Y$와 $Z$가 덧셈 독립(additive independence), 즉 $Y$와 $Z$는 상호 독립인지, 그리고 유틸리티 독립(utility independence)인지 확인한다. $Y$와 $Z$가 유틸리티 독립이라는 의미는 $Y$의 값의 변화에 따른 $Y$에 대한 선호도의 변화가 $Z$의 변화와는 관계가 없다는 의미이다.

**정성적인 성질 확인 :** 유틸리티 함수가 단조(monotonic) 함수인지 확인한다. $x_k$가 $x_j$보다 크면 $x_k$가 $x_j$보다 좋은 것인지 확인하고, 다음으로 유틸리티 함수 $u$가 모험 회피(risk averse), 모험 중립(risk neutral), 모험 노출(risk prone) 중에서 어떤 것인지 결정한다.

**유틸리티 함수 결정 :** 간단한 예로 의사 결정자의 유틸리티 함수가 $x$에서 단조증가이고, 감소적인 모험 회피라고 가정하면, 이러한 특징을 만족시키는 유틸리티 함수는 다음과 같다.

$$u(x) = h + k(-e^{-ax} - be^{-cx})$$

여기서 $a, b, c, k$는 양의 상수이다. 일반적으로 유틸리티 함수 $u(x_1, x_2, \cdots, x_n)$가 덧셈 독립, 유틸리티 독립이면 $u$는 다음과 같다.

$$u(x_1, x_2, \cdots, x_n) = k_1 u_1(x_1) + k_2 u_2(x_2) + \cdots + k_n u_n(x_n)$$

여기서 모든 $i$에 대해서 $u_i(x_i^0) = 0, u_i(x_i^*) = 1$이며 $u_i(x_i^0)$와 $u_i(x_i^*)$는 각각 가장 선호하지 않는 결과 유틸리티와 가장 선호하는 결과 유틸리티 이다.

MAUT는 복잡한 의사 결정 문제를 여러 개의 작고 단순한 문제로 나누어 보다 용이하게 해결할 수 있는 논리적 기틀을 제공한다. MAUT는 다양한 곳에 사용될 수 있으며 특히 전자 상거래 분야에서 판매자와 구매자 사이의 여러 Attribute에 대해 협상하는 과정에서 많이 사용되고 있다.

## 1.2 GRBAC (Generalized Role-Based Access Control)

지금까지 보안은 정적인 것으로 인식되어서 접근 제어도 컨텍스트에 따라 변경되지 않았으며 동적인 환경의 변화를 반영할 수 없었다. 이를 해결하기 위해서 상황인식 컴퓨팅을 보안 서비스에 도입하여야 한다. 상황인식 컴퓨팅을 위한 보안 서비스는 자연스럽고, 사용자의 개입을 최소로 하는 독립적인 형태의 것이어야 한다. 따라서 더 이상 보안을 위한 사용자 세션이 시관에 무관하게 동일한 인증 기법과 접근 기법을 사용한다고 가정할 수 없다[12]. 예를 들면, 상황인식 인증 기법은 풍부한 정보로 이루어져 있는 디지털 홈과 같은 환경에서는 굉장히 복잡하다. 인증 정책은 여러 요소에 의해서 제약을 받는데 사용자 종류와 사용자의 현재 위치 등의 정보에 따라서 많이 달라진다.

기존의 RBAC는 역할에만 기반을 두고 상황 정보를 사용하지 못했기 때문에 다음과 같은 한계를 가지게 되었다[13,14,15].

- 개별 사용자들이 멤버로서 요구되는 역할에 근거하여 이루어짐
- 접근 권한은 역할 이름에 따라 그룹화 됨
- 역할이 사용자 업무 책임과 권한에 따라 부과됨으로써 자원의 이용에 제약

이를 해결하기 위하여 GRBAC(Generalized Role-Based Access Control) [16,17] 개념을 고려하게 되었는데, 이는 기존의 역할기반 접근 제어가 시간에 따른 접근제어 등과 같이 상황에 근거한 접근제어를 수행할 수 없었던 문제점을 해결하고 있다. GRBAC는 상황에 근거한 접근제어를 수행하기 위하여, 접근제어 결정에 사용자 역할(subject role), 객체 역할(object role), 환경 역할(environment role)을 추가하여 기존 RBAC 모델을 확장하였다.

GRBAC 모델은 기존 RBAC 모델보다 상황인식 애플리케이션의 접근 정책을 나타내는데 강력하고 융통성있는 방법을 제공하고 있다[18]. 또한 디지털 홈 등을 실제 사용하게 될 사용자들은 컴퓨터나 보안 관련 초보자일 가능성이 크므로 여기에 사용될 애플리케이션에서 보안 정책을 쉽게 정의하고 사용할 수 있는 기능은 매우 중요하다. 이런 면에서  다양한 역할에 바탕을 둔 GRBAC 모델을 이용한 보안 서비스는 유비쿼터스 컴퓨팅 환경의 보안 서비스를 구현하는데 유용한 방법이 될 수 있다.

## 1.3 MAUT와 GRBAC를 이용한 상황인식 보안 시스템[7]

RFID/USN 환경은 언제 어디서나 사용자가 원하는 정보와 서비스를 제공하는 환경의 특성상 개인 프라이버시 침해 문제와 같은 많은 보안 위협을 가지고 있다. 그러나 RFID/USN 환경을 구성하는 장비들의 열악한 성능에 의해 많은 연산을 필요로 하는 현재의 보안 서비스는 제대로 적용되지 못하고 있다. 제안된 모델은 역할을 기반으로 사용자의 접근 권한을 제어하는 GRBAC 모델을 확장하고 그 결과를 사용자 선호도에 의한 가중치 조절로 의사를 결정하는 MAUT 모델에 적용하여 가장 적절한 보안 등급을 계산함으로써 결정된 보안 등급에 따라 유연한 보안 서비스를 적용하는 상황인식 보안서비스를 구현한다.

그러나 GRBAC 모델은 고정된 규칙을 기반으로 여러 가지 상황을 분류하여 권한을 부여하므로, 예기치 못한 상황에는 적절하게 대응하지 못한다는 단점을 가지며, 이를 해결하기 위해서는 모든 상황에 대한 규칙을 지정해야만 한다. 그러나 발생 가능한 모든 상황을 등록한다는 것은 불가능하다고 볼 수 있다. 또한 GRBAC 모델을 포함한 RBAC 계열의 모델은 다음과 같은 권한 할당에 대한 무결성 보장문제를 가진다.[19]

- 분리권한 (Disjoint Permission: DP) 제약조건: 동일한 권한이 정적인 임무 분리가 선언된 두 개 이상의 역할에 할당되지 못한다.
- 충돌 권한 (Conflict Permission: CP) 제약조건: 충돌 권한들은 동일 역할에 할당될 수 없다.
- 선행 권한 (Prerequisite Permission: PP) 제약조건: 권한 q가 권한 p의 선행 권한이라고 할 때, 권한 p가 임의의 역할에 할당되려면 권한 q가 이미 그 역할에 할당되어 있어야 한다.

- 단일 역할에만 권한 할당 (Permission Assignment to Single Role: PASR) 제약조건: 권한의 집합인 pasr_ps에 속한 권한들은 어떤 역할 pasr_r에만 할당될 수 있다.

이와 같은 단점을 해결하기 위해서 자율적인 분류 알고리즘을 이용하여 상황을 분류하고 계속 분류 기준을 갱신해 나가는 방법을 고려할 수 있다. 이러한 분류 알고리즘으로는 FCM (Fuzzy C-Means) 클러스터링 알고리즘 등이 있다.

## 2. FCM (Fuzzy C-Means) 클러스터링 알고리즘

FCM[20]은 1973년 Bezdek에 의해 제안된 알고리즘으로서 하나의 클러스터에 속해있는 각각의 데이터 점을 소속도에 의해서 분류하는 데이터 분류 알고리즘이다.

FCM은 n개의 벡터 $x_i(i=1,2,\cdots,n)$의 집합을 c개의 퍼지 그룹들로 분할하고, 각각의 그룹 안에서 클러스터의 중심을 찾는다. FCM 클러스터링 방법은 0과 1사이의 소속정도에 의해 주어진 데이터 점이 몇 개의 그룹에 속할 수 있다는 퍼지 분할 기법을 사용한다. 소속 함수 U는 0과 1사이의 값으로 구성되며 데이터 집합에 대한 소속정도 값의 합은 항상 1이다[21].

$$\sum_{i=1}^{c} u_{ik} = 1,\ \forall k = 1,\cdots,n \qquad 0 < \sum_{k=1}^{n} u_{ik} < n$$

FCM 클러스터링에 대한 비용함수는 다음과 같은 형태를 가진다.

$$J(u_{ik}, v_i) = \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^{m}(d_{ik})^2 \quad , \qquad v_i = \{v_{i1}, v_{i2}, \cdots, v_{ij}, \cdots, v_{iL}\}$$

$$d_{ik} = d(x_k - v_i) = \left[\sum_{j=1}^{1}(x_{kj} - v_{ij})^2\right]^{\frac{1}{2}}$$

$$v_{ij} = \frac{\sum_{k=1}^{n}(u_{ik})^m x_{kj}}{\sum_{k=1}^{n}(u_{ik})^m}, \quad (j = 1, 2, \cdots, L)$$

$$u_{ik} = \frac{1}{\sum_{j=1}^{c}\left(\frac{d_{ik}}{d_{jk}}\right)^{\frac{2}{m}-1}}$$

$u_{ik}$ : $i$번째 클러스터에 속한 $x_k$의 $k$번째 데이터의 소속 정도

$v_i$ : $i$번째 클러스터의 중심벡터

$m(1 \le m < \infty)$ : 소속 함수의 퍼지성의 정도에 대한 영향을 나타내는 지수의 가중치. 분류 공정에서 퍼지성의 양을 제어하는 파라미터. 일반적으로 $m = 2$

$d_{ik}$ : 클러스터에 속한 $k$번째 데이터 $x_k$와 $i$번째 클러스터의 중심벡터 $v_i$ 사이의 거리

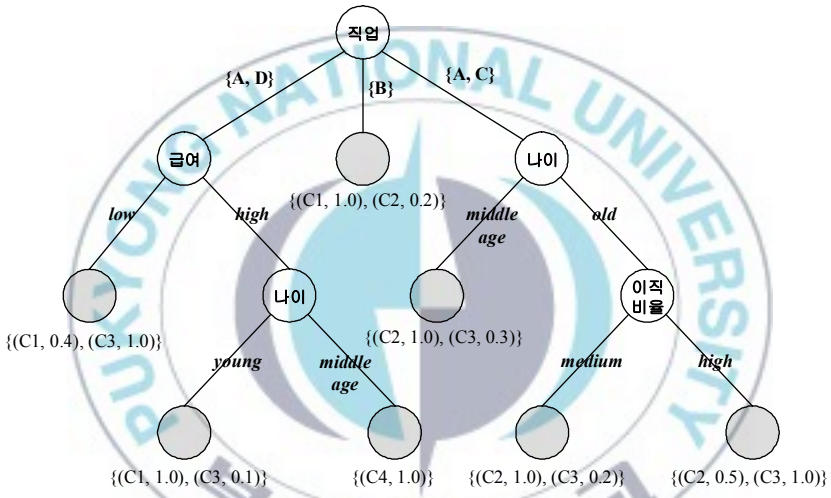
$J(u_{ik}, v_i)$ : FCM 클러스터링에 대한 비용함수

## 3. 퍼지 결정트리

결정트리는 분류규칙을 표현하는 트리이다. 비단말모드에는 분류를 위해 비교하는 데이터의 특징이 명시되고, 링크에는 비교조건 또는 특징 값이 부여되며, 단말노드에는 루트노드에서 해당 노드까지의 경로 상에 있는 모

든 조건을 만족하는 데이터가 속하는 클래스 값이 부여된다.

결정트리는 명확한 값을 기준으로 특징공간을 분할하므로, 미세한 차이를 가지는 서로 다른 두 데이터를 각각 다른 클래스로 분류할 수 있다. 따라서 결정트리에 퍼지함수의 개념을 도입하여, 특징공간을 소속함수를 이용하여 정의한 퍼지 경계면으로 분할하는 퍼지 결정트리에 대한 연구가 진행되고 있다[22]. 그림 2는 퍼지 결정트리의 예를 나타낸다[23].




**[그림 2] 퍼지 결정트리의 예**

그림 2에서 루트노드의 '직업'과 같은 비수치 속성의 링크에는 {A, D}와 같이 속성 값의 집합이 부여되고, '급여'같은 수치 속성의 링크에는 'high', 'low'와 같은 퍼지 언어항이 부여된다. 단말노드에는 루트노드로부터 해당 단말노드까지의 경로가 가지는 모든 조건을 만족하는 데이터가 속하는 클래스를 나타내며, {(C1, 0.4), (C3, 1.0)}과 같은 퍼지집합 형태로 표시된다. {(C1, 0.4), (C3, 0.1)}는 해당 단말노드에 도달하는 데이터는 클래스 C1에는 0.4, 클래스 C3에는 1.0의 가능성으로 소속됨을 의미한다[23].

# Ⅲ. FCM 클러스터링 알고리즘과 FDT를 적용한 상황인식 다중보안서비스 모델

기존의 모델은 확장된 GRBAC 모델을 기반으로 데이터를 분류하고 그 결과를 MAUT 모델에 적용하여 가장 적절한 보안 등급을 계산함으로써 결정된 보안 등급에 따라 유연한 보안 서비스를 적용하는 상황인식 보안서비스를 구현하였다. 그러나 RBAC 계열의 모델은 보안정책에 등록되지 않는 상황에는 대응하지 못한다는 단점을 가진다.

이러한 단점을 해결하기 위하여 본 논문에서는 GRBAC와 MAUT를 결합한 모델 대신 FCM 클러스터링 알고리즘과 퍼지 결정트리 (Fuzzy Decision Tree : FDT)를 적용한 상황인식 보안 서비스 모델을 제안한다.

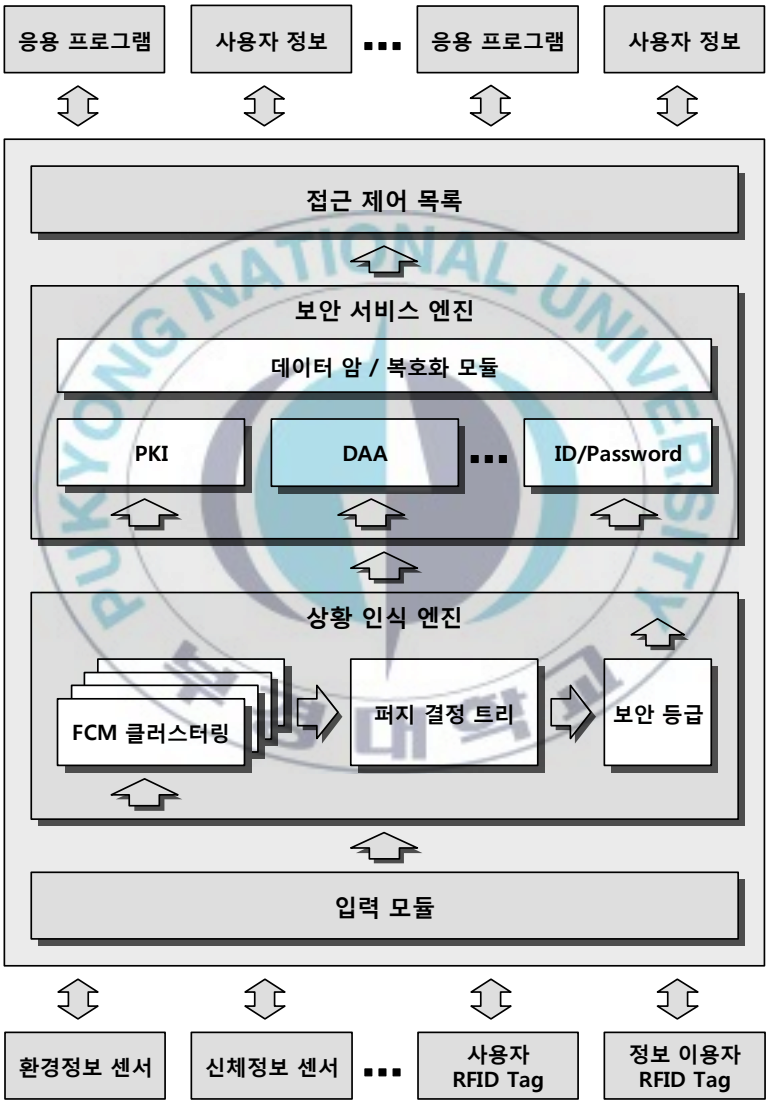
## 1. 제안 모델의 구조

본 논문에서는 FCM 클러스터링 알고리즘과 퍼지 결정트리 (Fuzzy Decision Tree : FDT)를 적용한 상황인식 보안 서비스 모델을 제안한다.

FCM 클러스터링 알고리즘을 이용한 상황인식 보안 서비스 모델은 분류된 각 클러스터의 중심과의 거리를 이용하여 가장 가까운 분류결과를 적용하므로 예기치 못한 상황에서도 가장 적절한 결과를 도출할 수 있다. 또한 최종적으로 도출된 보안 등급에 대한 접근제어목록(Access Control List : ACL)을 적용할 경우 다양한 상황의 충돌에 대한 무결성 보장문제도 해결

된다.

본 논문에서 제안하는 모델은 입력모듈, 상황인식 엔진, 보안 서비스 엔진, 접근 제어 목록으로 구성된다. 그림 3은 제안 모델의 구조를 나타낸다.

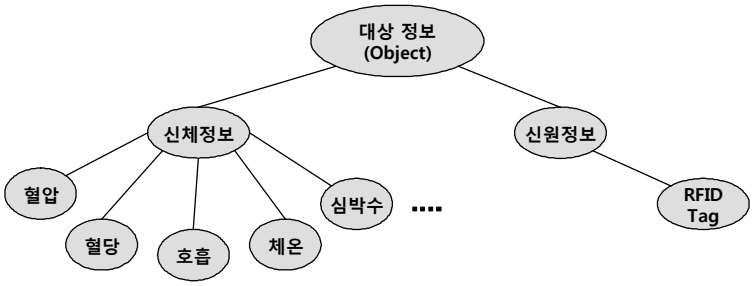



[그림 3] 제안 모델의 아키텍처

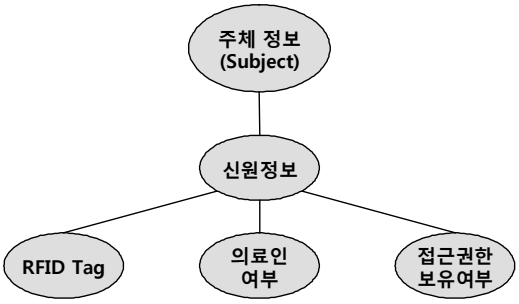
## 2. 입력 모듈

입력 모듈은 센서네트워크의 다양한 센서들이 제공하는 상황정보를 수집하는 역할을 수행한다. 센서 네트워크는 온도, 습도, 조도 등의 환경정보 센서 및 체온, 혈압, 맥박수와 같은 인체정보 센서, 그리고 진동, 소음상태, 가스누출 여부 등을 알려주는 다양한 센서로 구성되어 있다. 입력모듈에서 수집한 데이터는 상황 인식 엔진으로 전달된다. 그림 4, 5, 6은 센서를 통해 입력받을 수 있는 상황정보의 구조를 나타낸다. 실제 입력되는 데이터는 각각의 센서로부터 전달받는 수치 데이터이며 입력모듈에서 시스템에서 사용가능한 데이터의 형태로 변환한다.


[그림 4] 환경정보의 구조의 예




[그림 5] 정보를 이용하고자 하는 대상 정보의 구조의 예




[그림 6] 정보를 이용하고자 하는 주체정보

# 3. 상황 인식 엔진

상황 인식 엔진은 FCM 클러스터링 알고리즘과 퍼지 결정트리를 이용하여 데이터를 분류하고 통합하여 사용자가 처한 상황을 인식하는 역할을 수행한다. 입력 모듈로부터 수집 및 전달되는 입력 데이터는 다양한 센서로부터 수집된 정보로 구성되며, 1회에 입력되는 입력 데이터는 센서의 종류만큼 존재한다. 따라서 입력 데이터는 센서의 종류에 따라 각각의 FCM 클러스터링 결과를 만들어 낸다. 즉 각각의 센서별로 보안 등급이 계산된다.

## 3.1 FCM 클러스터링

분류는 데이터를 기존에 설정된 클래스와 비교하여 가장 근접한 클래스로 판별하는 것이며 패턴인식, 의사결정, 데이터 분석 등에서 가장 핵심적인 작업이다. 시스템이 요구하는 정보 처리량이 방대해지면서 주어진 정보를 분석하여 다양한 상황에 대응하는 기능은 지능형 시스템의 기본 기능이

되었고, 이러한 적응기능을 위하여 신경망을 비롯한 다양한 알고리즘이 제안되었다. 신경망의 연구는 학습을 통한 분류 기법의 대표적인 방법이며 특히 SOM의 경우 스스로 비슷한 속성의 데이터끼리 모이도록 학습함으로써 클러스터 분석을 통한 분류에 활용되어 왔다. 그러나 승자독점의 방식을 이용하여 특정 뉴런의 가중치를 갱신하면서 학습이 진행되므로, 실세계에서의 경계가 불명확한 데이터에 대해서는 파리미터에 의존적인 결과를 가지게 되었다. 이에 퍼지 이론을 적용하여 각 데이터의 클래스 소속 값을 함께 학습함으로써 더욱 정확한 분류 결과를 얻을 수 있는 퍼지 클러스터링 기법이 도입되었다. FCM은 대표적인 클러스터 분석방법의 하나로서 여러 응용 분야에 적용되었으며, 아직 이 방법의 수렴성과 최적화, 일반화에 대한 고찰이 진행 중인 알고리즘이다[24].

FCM 클러스터링 모듈에서는 각 센서정보에 따라 소속 함수를 설계하고 소속정도를 계산한다. 계산된 소속정도를 이용하여 각 정보에 대한 보안등급을 도출한다. 표 2는 FCM 클러스터링 알고리즘을 나타낸다[21].

## [표 5] FCM 클러스터링 알고리즘

[단계 1] 클러스터의 개수 $c\,(2 \le c \le n)$ 결정
지수의 가중치 (exponential weight) $m\,(1 < m < \infty)$ 선택
초기 소속 함수 $U^{(0)}$ 초기화
알고리즘 반복회수를 $r\,(r = 0, 1, 2, \cdots)$로 설정

[단계 2] 퍼지 클러스터의 중심 $\{v_i | i = 1, 2, \cdots, c\}$을 계산

$$v_{ij} = \frac{\sum_{k=1}^{n} (u_{ik})^m x_{kj}}{\sum_{k=1}^{n} (u_{ik})^m}$$

[단계 3] 새로운 소속 함수 $U^{(r+1)}$을 계산

$$u_{ik}^{(r+1)} = \frac{1}{\sum_{j=1}^{c} \left(\frac{d_{ik}^r}{d_{jk}^r}\right)^{\frac{2}{m}-1}} \quad \text{for } I_k = \Phi \quad , \quad \text{또는}$$

$$u_{ik}^{(r+1)} = 0 \quad \text{for } all\ classes\ i,\ i \in \tilde{I}_k$$

$$I_k = \left\{ i | 2 \le c < n; d_{ik}^{(r)} = 0 \right\},\ \tilde{I}_k = \{1, 2, \cdots, c\} - I_k$$

$$\sum_{i \in I_k} u_{ik}^{(r+1)} = 1$$

[단계 4] 식 A를 계산하여 $\Delta > \epsilon$ 이면 $r = r+1$로 정하고 [단계 2]부터 반복 수행, $\Delta \le \epsilon$ 이면 알고리즘 종료. ($\epsilon$ 는 임계값)

$$\Delta = \| U^{(r+1)} - U^{(r)} \| = \max_{i,k} \left| u_{ik}^{(r+1)} - u_{ik}^{(r)} \right|$$

## 3.2 퍼지 결정 트리

결정트리는 분류 또는 의사결정을 위해 특징 속성 값으로 표현된 사례들로부터 분류지식을 추출하기 위해 널리 사용되어온 방법이며, 표준 패턴을 자동으로 추출하기 위한 방법론에는 결정트리생성, 신경회로망 모델, 확률기반 모델, 진화연산 기법 등이 있다. 결정트리는 단변수 결정트리와 다변수 결정트리로 나뉜다. 특히 의사 결정트리는 원인을 규칙으로 표현할 수 있기 때문에 사용자가 원인을 쉽게 이해할 수 있고 시스템의 구축을 용이하게 한다는 장점을 가진다. 실세계의 데이터는 관측오류, 불확실성, 주관적인 판단 등으로 인해서 애매한 형태로 주어지며 대부분의 기존 결정트리 생성방법은 데이터에 대포된 애매함에 대해 충분히 고려하지 못하고 있다. 또한 일반 결정트리는 명확한 값을 기준으로 특징공간을 분할하기 때문에 미세한 차이를 가지는 두 데이터를 서로 다른 클래스로 분류할 수 있다. 이에 특징공간에 분명한 분류 경계를 설정하는 대신에 퍼지 경계를 설정하는 방식을 이용하는 퍼지 결정트리가 제안되었다[22,23].

제안 모델은 최종적으로 가장 적절한 하나의 보안 등급을 필요로 한다. 따라서 FCM 클러스터링을 통해 도출된 각각의 보안 등급은 퍼지 결정트리를 이용하여 최종적으로 하나의 보안 등급으로 통합된다. 입력 데이터를 이루는 각 센서 정보들은 센서의 종류에 따라 서로 다른 중요도를 가진다. 예를 들면 고혈압 환자의 신체정보 중에서 혈압과 심박 수가 있을 때, 심박 수의 정보보다 혈압의 정보가 더욱 중요한 정보인 것과 같다. 따라서 제안 모델에서는 FCM 클러스터링을 통해 도출된 각각의 보안 등급에 중요도를 곱한 값을 퍼지 결정트리의 입력 데이터로 사용한다.

제안 모델은 다양한 센서의 정보를 이용하므로 FCM 클러스터링을 통해서 제시되는 입력 데이터도 다양한 변수로서 적용되며, 본 모델에서는 다

변량 개념을 퍼지 결정트리에 적용한 다변량 퍼지 결정트리 (Multivariate Fuzzy Decision Tree : MFDT)[25]를 이용하여 최종 보안 등급을 도출한다. 다변량 퍼지 결정트리의 학습은 표 3과 같은 과정을 따른다 [25].

**[표 6] 다변량 퍼지 결정트리 학습**

[단계 1] 근노드를 생성하고 모든 학습데이터 $x$를 근노드에 위치시킨다.

[단계 2] 정보이득을 최대화하는 노드를 생성

1) LDA (Linear Discriminant Analysis)를 이용하여 속성벡터 w 결정

$$Maximize\, J(w) = \frac{w^T S_B w}{w^T S_w w}$$

$$S_w = \sum_{i=1}^{k} S_i \ , \quad S_i = \sum_{x \in class\, i} (x - m_i)(x - m_i)^T$$

$$m = \frac{1}{K} \sum_{i=1}^{K} m_i$$

2) $w$를 이용하여 속성 값 $z = w^T x$ 계산

3) 현재 노드의 엔트로피 계산

$$Entropy(S) = -\sum_i P_i^S \log_2 P_i^S$$

$$P_i^S = \frac{N_S^i}{N_S}, \ N_S = \sum_i N_S^i$$

$N_S^i$ : $i$번째 클래스인 데이터의 개수

$S$ : 현재 노드에 도달한 $z$의 집합

4) 노드분기 시 정보이득을 최대화 하는 소속함수 및 정보이득을 계산

$$m^{v,c} = \frac{1}{n_v} \sum_{j=1}^{n_v} z_j^v \ , \quad v = 1, \cdots, N$$

$n_v$ : $v$번째 구역에 포함되는 데이터 $z$의 개수

$z_j^v$ : $z$를 오름차순으로 정렬했을 때 $j$번째 $z$값

$m^{v,c}$ : $v$번째 소속 함수의 꼭지점 위치

**[표 3] 다변량 퍼지 결정트리 학습(계속)**

모든 구간에서 각 소속 함수의 꼭지점 위치를 구한 후 소속 함수의 좌측, 우측 값을 계산

$$m^{v,l} = m^{v,c} - 0.5(1+\gamma)(m^{v,c} - m^{v-1,c})$$
$$m^{v,r} = m^{v,c} + 0.5(1+\gamma)(m^{v+1,c} - m^{v,c})$$

$\gamma$ : 두 퍼지 소속 함수 간의 겹친 정도

$$S_{v|w} = \{(z, \mu_{S_{v|w}}(z)) \mid \mu_{S_{v|w}}(z) : v\text{번째 소속 함수의 소속 값}\}$$

$$C^i_{S_{v|w}} = \sum_{\substack{class\ of\ z = i \\ z \in P(S_{v|w})}} \mu_{S_{v|w}}(z)$$

$$C_{S_{v|w}} = \sum_i C^i_{S_{v|w}}, \quad P_i^{S_{v|w}} = \frac{C^i_{S_{v|w}}}{C_{S_{v|w}}}$$

$$Entropy(S_{v|w}) = -\sum_i P_i^{S_{v|w}} \log_2 P_i^{S_{v|w}}$$

$$Gain(S, w) = Entropy(S) - \sum_v \frac{N_{S_{v|w}}}{N_S} Entropy(S_{v|w})$$

5) 현재 노드를 한 가지 속성만 사용해서 분기하는 경우, 가장 큰 정보 이득을 갖게 하는 속성 및 소속 함수를 구한다. ($w$는 하나의 요소만 1이고 나머지는 0인 단위 벡터가 됨)

6) 다변량 속성벡터를 사용한 분기 (4번 과정)와 하나의 속성을 사용한 분기 (5번 과정)의 정보 이득 중 더 큰 정보 이득을 갖는 $w$와 소속 함수를 사용해 자식 노드를 생성한다.

[단계 3] 종료조건을 만족하면 현재 노드를 단말 노드로 만들고 클래스 할당. 그렇지 않으면 모든 자식 노드에서 단계 2를 재귀적으로 반복

종료조건 :

1) 현재 노드의 모든 데이터의 클래스가 동일한 경우
2) 현재 노드의 깊이가 미리 정의된 최대 깊이를 초과하는 경우

표 4는 다변량 퍼지 결정트리를 이용한 분류과정을 나타낸다[25].

**[표 4] 다변량 퍼지 결정트리를 이용한 분류과정**

[단계 1] 근 노드에서 각 단말 노드까지 거쳐가는 노드에서의 소속 함수와 속성 벡터를 이용하여 T-norm (Triangular-norm)을 계산

1) 각 노드의 속성 벡터 $w_i$의 방향으로 입력데이터 $x$를 사영시킨다.

$$z_i = w_i^T x$$

2) 근 노드에서 $n$번째 단말 노드까지의 T-norm을 계산

$$T_n = \prod_{i=\sqrt{node}}^{n_{th} \le af\,node} \mu_{S_{th}w}^{i}(Z_i)$$

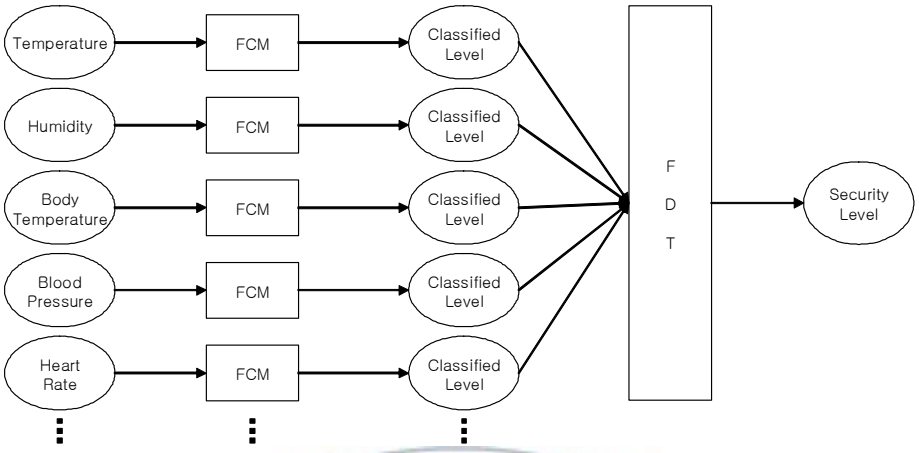
위 과정을 모든 단말 노드에 대하여 반복하면 각 단말 노드는 0에서 1 사이의 T-norm을 가짐.

[단계 2] 단말 노드 중에 각 클래스에 해당하는 단말 노드들의 T-norm의 평균을 계산하고 이 평균 값이 가장 큰 클래스로 입력 데이터를 분류한다.

$A_i$ : $i$번째 클래스를 갖는 단말 노드들의 T-norm의 평균

$$class\ C = \arg_i \max A_i$$

최종 보안등급에 따라 보안서비스 엔진에서 제공하는 다양한 보안기능을 수행하고 ACL에서 제공되는 접근권한에 따라 정보에 접근하게 된다. 그림 7은 제안 모델의 보안 등급 도출 과정을 나타낸다.




[그림 7] 보안등급 도출 과정

# 4. 보안 서비스 엔진

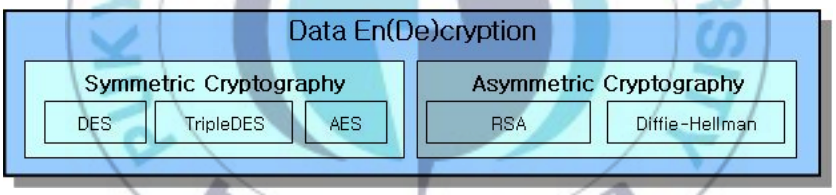
보안 서비스 엔진은 정보에 대한 기밀성을 유지하기 위한 다양한 데이터 암/복호화 모듈과 사용자의 신분을 인증하기 위한 인증 모듈로 구성된다. 데이터 암/복호화 모듈에서는 대칭키 및 비 대칭키 암/복호화 모듈을 지원하며 인증모듈에서는 간단한 ID/Password 모듈부터 PKI (Public Key Infrastructure) 모듈, DAA (Direct Anonymous Attestation) 모듈 등 다양한 인증모듈을 제공한다. 이러한 모듈을 이용하여 도출된 보안 등급에 따라 적절한 보안 서비스를 제공하게 된다.

## 4.1 데이터 암/복호화 모듈

보안서비스 엔진의 데이터 암/복호화 모듈에서는 데이터의 기밀성, 무결

성 제공을 위해 키 일치 방법을 사용한다. 대칭키 방법은 데이터를 암호화하는 시간은 짧은 반면 데이터를 주고받는 상호간의 비밀 키에 대한 비밀 유지가 어렵다. 그리고 비 대칭키 방법은 암호화하는데 많은 시간이 걸려 효율적이지 못한 문제점을 지니고 있다. 키 일치 방식은 이러한 단점들을 피하고 대칭키, 비 대칭키 방식의 장점만을 이용하여 안전하고 빠른 데이터 송수신이 가능하게 한다[26,27].

키 일치에는 Diffie-Hellman 알고리즘을 사용한다. 또한 개발자가 다른 암호화 방법을 사용하고자 할 때를 대비하여, TripleDES, AES, RSA와 같은 여러 암호화 방법을 사용할 수 있는 API를 지원한다. 지원하는 API를 이용하여 새롭게 구현한 알고리즘을 암/복호화 모듈에 추가 및 사용할 수 있다. 그림 8은 데이터 암/복호화에 대한 아키텍처를 나타낸다[28].




[그림 8] 데이터 암/복호화 아키텍처

표 5는 대칭키 암호화 관련 클래스의 이름과 기능에 대한 설명을 보여주며, 표 6은 비대칭키 암호화 관련 클래스의 이름과 기능에 대한 설명을 나타낸다.

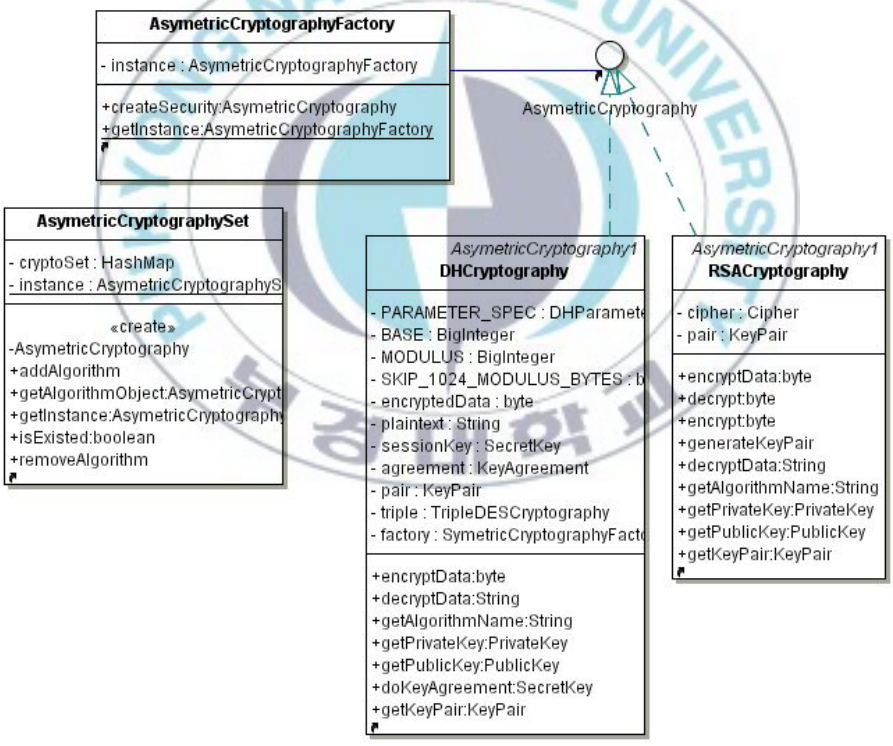
[표 5] 대칭키 암호화 관련 클래스 설명

| 클래스 이름 | 기능 |
|---|---|
| SymmetricCryptography | 대칭 암호화 기능(DES, TripleDES, AES)을 제공하는 product 클래스의 인터페이스 |
| DESCryptography | DES 암복호화 기능을 제공하는 Cryptography product 클래스 |
| TripleDESCryptography | TripleDES 암복호화 기능을 제공하는 Cryptography product 클래스 |
| AESCryptography | AES 암복호화 기능을 제공하는 Cryptography product 클래스 |
| SymmetricCryptographyFactory | 대칭키 암호화 product 클래스들을 개발자가 생성할 수 있게 도와주는 factory 클래스 |
| SymmetricCryptographySet | 대칭키 암호화 product 클래스를 관리하고, 새로운 비대칭 암호화 product 클래스를 추가/제거 하는 기능을 가진 클래스 |

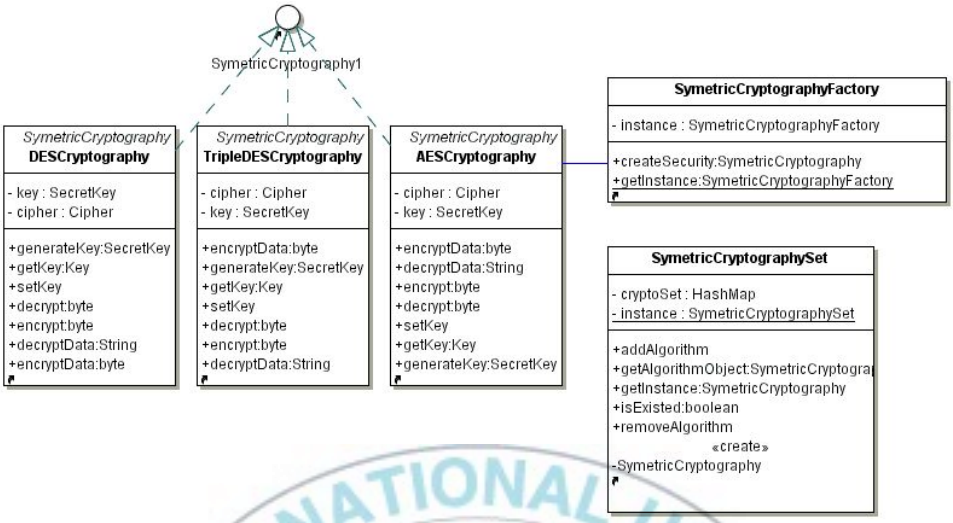
[표 6] 비대칭키 암호화 관련 클래스 설명

| 클래스 이름 | 기능 |
|---|---|
| AsymmetricCryptography | 비대칭 암호화 기능(RSA, DH)을 제공하는 클래스의 인터페이스 |
| RSACryptography | RSA 암복호화 기능을 제공하는 Cryptography product 클래스 |
| DHCryptography | DH 암복호화 기능을 제공하는 Cryptography product 클래스 |
| AsymmetricCryptographyFactory | 비대칭키 암호화 product 클래스들을 개발자가 생성할 수 있게 도와주는 factory 클래스 |
| AsymmetricCryptographySet | 비대칭키 암호화 product 클래스를 관리하고, 새로운 비대칭 암호화 product 클래스를 추가/제거 하는 기능을 가진 클래스 |

암호화의 방법이 대칭키, 비대칭키로 구분되기 때문에 클래스가 크게 두 부분으로 나뉘어져 있다. 데이터 암/복호화 클래스들은 Factory 패턴을 적용하며 이러한 패턴의 적용으로 인해 개발자는 암호화 관련 클래스가 어떤 식으로 구성되는지 알지 못하더라도 쉽게 애플리케이션을 구현할 수 있으며 유지보수가 쉬운 장점을 가진다. 그림 9는 비대칭 암호화 관련 클래스의 구조를 보여주며, 그림 10은 대칭 암호화 관련 클래스의 구조를 보여준다.




[그림 9] 비대칭 암호화 관련 클래스




[그림 10] 대칭 암호화 관련 클래스

## 4.2 인증 모듈

인증이란 자신의 신분과 행위를 증명하는 것을 말하며 사이버 공간에서 구현되는 어떤 서비스에 대해서도 가장 기본적이고 중요한 보안개념이라고 할 수 있다. 인증 방법에는 사용자 ID와 비밀번호를 이용하는 ID/Password 방식과 공개키 기반 구조 (PKI) 방식, 장비 간의 인증을 수행하는 DAA (Direct Anonymous Attestation)[29] 방식 등이 있다.

### 4.2.1 ID/Password 방식

컴퓨터 또는 네트워크에 로그인 하는 사용자 개인을 식별하는데 사용되는 문자집합인 ID와 식별된 사용자의 신원을 확인하는데 사용하는 비밀번호 집합으로 인증이 이루어지는 방식이다.

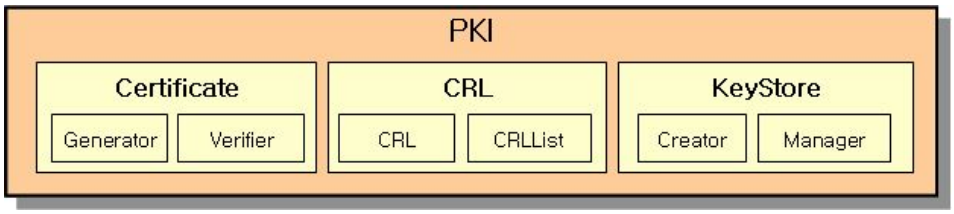
ID/Password 방식은 구현이 쉽고 인증속도가 빠르다는 장점이 있는 반

면, 평문으로 구성된 비밀번호를 사용하면 통신상에 노출될 위험이 크고, 인간의 기억에 의존하므로 망각하기 쉬우며, 비밀번호 추측 공격(Password Guessing Attack)에 취약하다는 등의 다양한 보안 취약점이 존재한다. 일반적으로 가장 많이 사용되는 방식의 하나이다.

### 4.2.2 PKI (Public Key Infrastructure)

PKI는 인증기관 (CA)으로부터 공개키와 개인키를 이용하여 생성된 인증서를 발급받아 네트워크상에서의 안전한 비밀 통신을 가능하게 해 주는 공개키 기반 구조로서 공신력 있는 인증기관에 의한 사용 주체의 인증, 사용 정보의 무결성과 기밀성, 부인방지의 기능을 담당한다. PKI는 사용자를 식별할 수 있는 디지털 인증서와 인증서를 저장했다가 필요할 때 불러 쓸 수 있는 디렉토리 서비스를 제공한다.

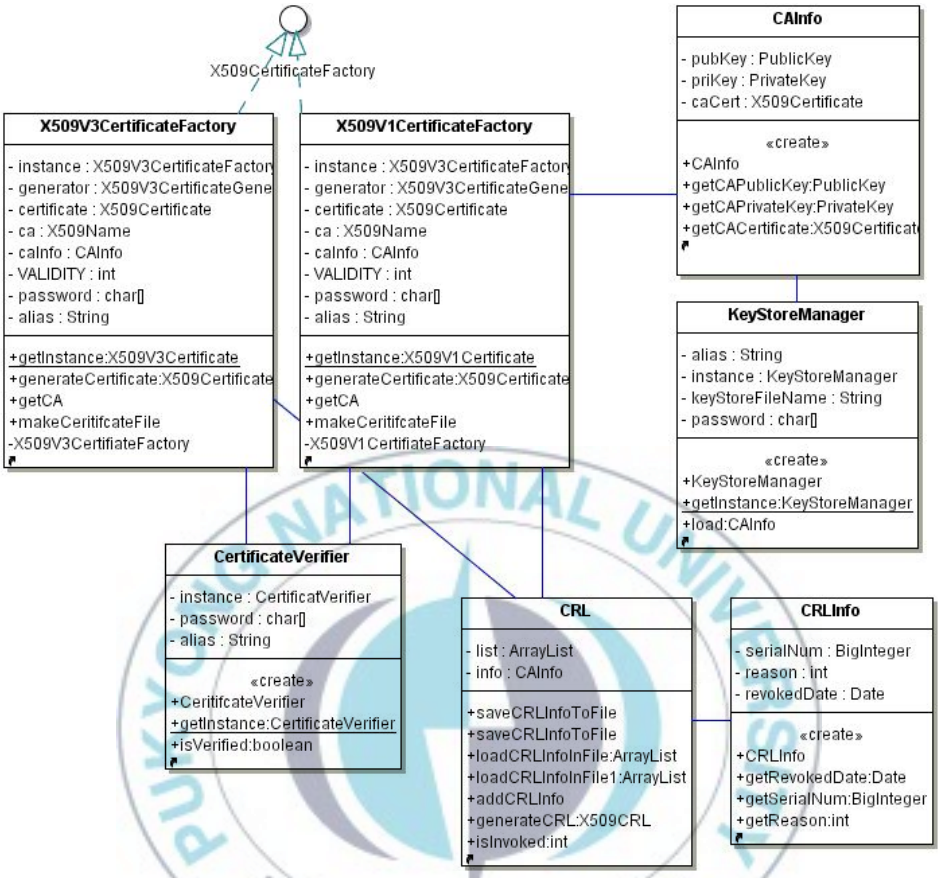
제안된 모델에서의 PKI 모듈은 X509. CertificateFactory를 이용하여 X509 버전1, 3에 대한 인증서를 생성한다. 또한 CertificateVerifier와 CRL을 통해서 인증서의 검증 서비스를 제공한다. 그림 11은 PKI 처리에 대한 아키텍처를 나타낸다[28]. 표 7은 X509 인증서와 관련된 클래스의 이름과 설명을 보여주며 그림 12는 PKI 환경 구성을 위한 클래스의 구조를 나타낸다.




[그림 11] PKI 아키텍처

[표 7] X509 인증서와 관련된 클래스 설명

| 클래스 이름 | 기능 |
|---|---|
| X509CertificateFactoryIF | 인증서를 생성하고, 검증하는 역할을 하는 Factory 클래스의 인터페이스 |
| X509V1CertificateFactory | X.509 버전 1의 인증서를 생성하고, 검증하는 역할을 하는 factory 클래스 |
| X509V3CertificateFactory | X.509 버전 3의 인증서를 생성하고, 검증하는 역할을 하는 factory 클래스 |
| CertificateVerifier | 클라이언트로부터 전송된 인증서를 실질적으로 검증하여 결과를 넘겨주는 클래스 |
| KeyStoreManager | 서버에 저장된 KeyStore를 이용하여 CA의 정보를 읽어오는 역할을 하는 클래스 |
| CAInfo | KeyStoreManager를 통해서 읽히는 CA정보를 담는 정보 클래스 |
| CRL | 사용자의 실수나 요청으로 인증서를 폐기해야 할 때 인증서 폐기 목록을 저장하고 처리하는 클래스 |
| CRLInfo | CRL에 저장된 폐기정보들 중 CA가 요구하는 정보만을 담는 정보 클래스 |
| CRLReason | 인증서가 폐기되는 이유 정보를 담은 정보 클래스로 CRLInfo에서 사용된다. |
| CRLResultInfo | CA의 요구로 생성되는 CRL 관련 결과 정보 클래스로 CA가 인증서의 폐기 여부를 CRL 클래스로 물어볼 때 만들어진다. 이 클래스는 인증서가 폐기되었는지 여부와 폐기되었으면 폐기된 이유를 가지고 있다. |



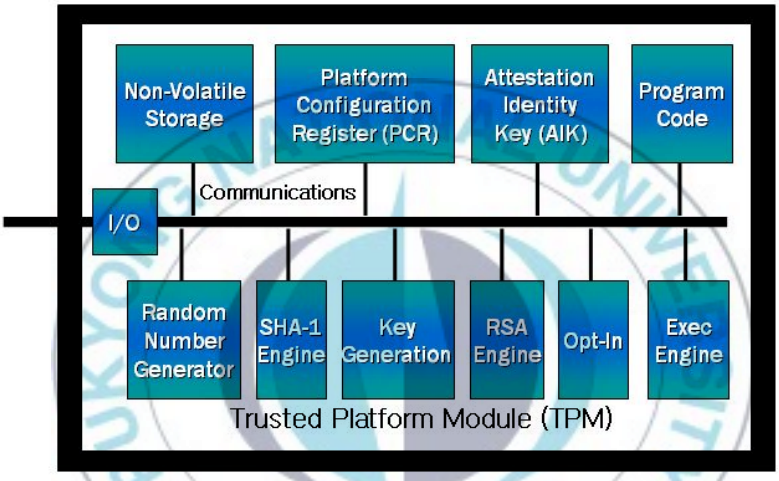

[그림 12] PKI 환경 구성을 위한 클래스

### 4.2.3 DAA (Direct Anonymous Attestation)

DAA는 TCG에서 설계된 영지식 증명에 기반을 둔 그룹 서명 스킴의 하나로 플랫폼 사용자에 대한 프라이버시를 보호하고 동시에 TPM 하드웨어의 원격 인증을 제공하는 서명 기법이다[29].

TCG는 이종 컴퓨터 플랫폼에서 컴퓨팅환경의 보안 강화를 목적으로 설립된 비영리 산업표준화 기관으로, TPM이라는 플랫폼 모듈을 정의하고

있다.[30] TPM은 플랫폼에 장착되어 보안과 관련된 정보를 저장하고 사용하며, 하드웨어 무결성을 보장해주는 플랫폼 모듈이다. TPM은 외부 소프트웨어 공격이나 물리적 공격에 대해 저장된 정보를 안전하게 보호하고, 암호 모듈과 난수 발생기 등을 사용한다. TPM은 그림 13과 같은 구성을 이루고 있다.




[그림 13] TPM의 구성

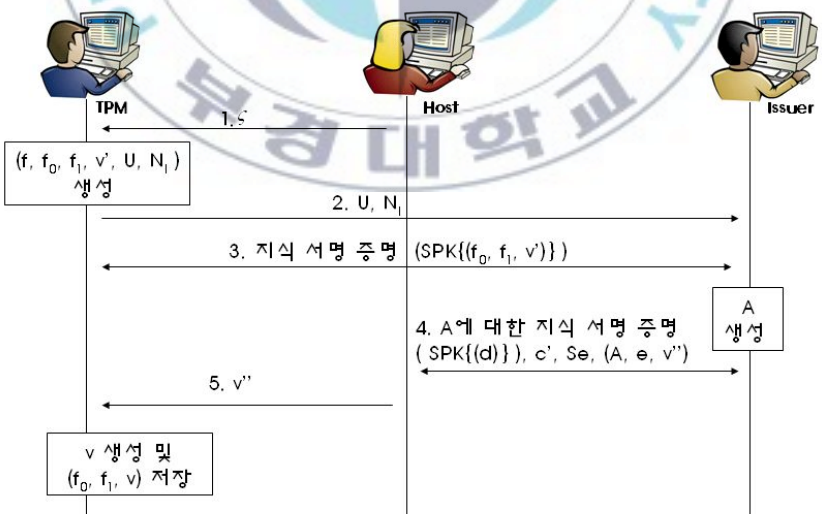
DAA는 TPM이 가지는 하나의 인증방식으로 TPM은 플랫폼의 인증과 신뢰 사슬을 이용하여 검증하는 인증 방식을 수행하며 사용자 및 디바이스에 관련된 정보를 전혀 제공하지 않으면서 자신의 타당성과 신뢰성을 인증할 수 있는 영지식 증명 기법을 이용한 DAA 기술을 지원한다.

DAA는 TTP(Trusted Third Party) 없이 인증 가능하고, 익명성을 제공하며, 불법 사용자(rogue TPM)를 찾는 방법이 존재한다. 그리고 Strong RSA 및 Decisional Diffie- Hellman Assumption을 사용하여 랜덤 오라클 모델에서 안전하다는 특징을 가지고 있다 [31].

DAA 프로토콜의 구성요소는 인증을 받기 원하는 TPM 사용자, 인증서 발급 기능을 하는 Issuer, TPM 사용자를 검증하는 Verifier로 구성되며 DAA의 인증 과정은 Setup, Join, Sign, Verification 과정을 거치게 된다 [32].

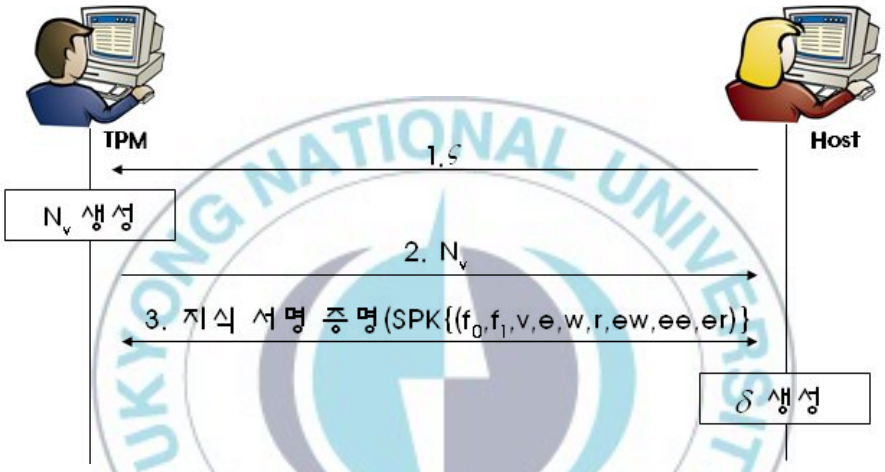
Setup Protocol은 Issuer의 공개키, 비밀키 쌍을 생성하고 공개키를 하위 노드에게 전송하는 과정이다. Issuer는 n=p*q를 만족하는 p, q와 이차잉여 그룹에 대한 랜덤 생성자 g′를 선택하여 공개키를 생성하고, p′=2p+1, q′=2q+1를 만족하는 p′*q′를 비밀키로 저장한 후 공개키를 하위 노드에게 전송한다.

Join 프로토콜은 노드들이 DAA 프로토콜을 이용하기 위한 인증서를 발급 받는 과정이다. 인증서를 발급 받기 위해서 노드는 TPM에서 생성한 비밀 정보를 가지고 있다는 것을 영지식 증명을 통해서 증명한다. Join 프로토콜의 처리과정은 그림 14와 같다.




[그림 14] Join 프로토콜 처리과정

Sign Protocol은 TPM를 소유한 노드가 메시지를 서명하고자 할 때 Join 프로토콜에서 얻은 비밀정보와 Issuer의 공개키, 서명될 메시지를 이용하여 전자서명을 생성하는 과정이다. Sign 과정에서는  자신의 인증서, Issuer의 공개키, 그리고 메시지를 이용하여 전자서명을 만든다. 그림 15는 Sign 프로토콜의 처리과정을 나타낸다.




[그림 15] Sign 프로토콜 처리과정

Verify Protocol은 전자서명에서 공개된 값들과 Issuer의 공개키를 이용하여 영지식 증명과 자신의 유효성을 판단하기 위한 정보를 생성 및 계산하여 전송받은 전자서명의 값과 일치하는지 비교한다.

# 5. 접근 제어 목록

접근 제어 목록은 분류 모듈을 통해 도출된 보안 등급에 따라 사용자가 접근 가능한 권한의 목록을 제공한다. 일반적으로 GRBAC와 같은 역할 기

반 접근제어 기법이 많이 사용되고 있으나 역할 기반 접근제어 기법은 예상하지 못한 상황, 즉 보안 정책에 설정되지 않은 문제에 대하여 대응이 어렵다는 단점을 가지며, 또한 분류를 위한 데이터의 종류가 늘어남에 따라 그 구조가 복잡해지고, 각각의 상황정보에 대한 조건에 따른 접근권한의 충돌문제로 인해 무결성을 유지하지 못하는 경우가 발생한다.

제안 모델에서는 복잡한 구조를 피함으로써 관리의 편의성을 향상시키고 복잡한 조건에 따른 충돌문제를 예방하기 위해 FCM과 FDT를 이용하여 최종적인 보안 등급을 계산하고, 이 보안 등급과 정보를 이용자의 권한 등급만을 이용하여 접근제어를 수행한다. 상황인식 모듈을 통하여 계산된 보안 등급을 기반으로 접근 제어를 수행하므로 이러한 접근권한 충돌 문제가 발생하지 않는다. 또한 접근 제어 목록의 내용을 관리함으로써 시스템 및 사용자의 정보에 대한 유연한 접근이 가능하도록 설정이 가능하며, 유사한 특성을 가진 다른 응용 프로그램에도 적용이 가능하다. 그림은 접근제어 목록의 예를 나타낸다. 도출된 보안 등급과 정보를 사용하고자 하는 사용자의 등급을 이용하여 허용 가능한 정보의 목록을 관리한다. 그림 16은 접근 제어 목록의 예를 보여준다.

```
<ACL>
	<POLICY>
		<SECURITY_LEVEL>1</SECURITY_LEVEL>
		<SUBJECT_LEVEL>1</SUBJECT_LEVEL>
		<OPERATION>Read</OPERATION>
		<LIST>
			<INFORMATION>Blood Pressure</INFORMATION>
			<INFORMATION>Heart Rate</INFORMATION>
			<INFORMATION>History of a  patient</INFORMATION>
			<INFORMATION>Clinical history</INFORMATION>
			<INFORMATION>Case history</INFORMATION>
			<INFORMATION>Special Attention</INFORMATION>
		</LIST>
	</POLICY>

	<POLICY>
		<SECURITY_LEVEL>2</SECURITY_LEVEL>
		<SUBJECT_LEVEL>1</SUBJECT_LEVEL>
		<OPERATION>Read</OPERATION>
		<LIST>
			<INFORMATION>Blood Pressure</INFORMATION>
			<INFORMATION>Heart Rate</INFORMATION>
			<INFORMATION>Case history</INFORMATION>
			<INFORMATION>Special Attention</INFORMATION>
		</LIST>
	</POLICY>
</ACL>
```

[그림 16] 접근 제어 목록 (ACL)의 예

# Ⅳ. 구현 및 평가

## 1. 구현

본 논문에서 제안한 모델의 학습 및 성능을 파악하기 위하여 Intel Pentium-IV 3.0GHz CPU와 1GB Ram이 장착된 PC에서 Java 5.0을 이용하여 구현하였다. 상황 정보의 입력을 위한 센서 장비로는 TynyOS 2.x를 기반으로 하는 UBee430/UBee430-AP-Kit을 사용하였다.

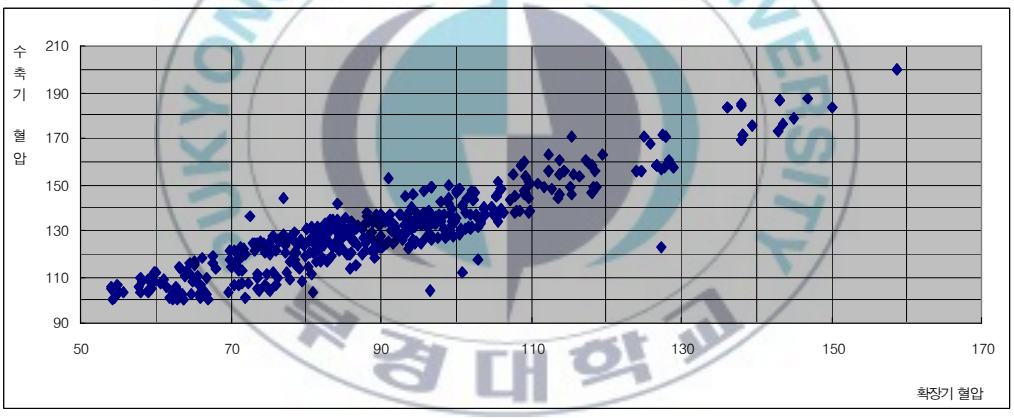
실험 데이터는 한국인 20세 이상의 성인을 대상으로 조사된 고혈압 유병률과 고혈압 진단 기준에 맞추어 생성한 300건의 가상 데이터를 사용하였다. 표 8은 정상 혈압과 고혈압의 진단 기준을 나타내며 표 9는 20세 이상의 한국인 성인에 대한 고혈압 유병률을 나타낸다[33]. 그림 17은 표 8과 표 9의 기준에 맞추어 생성한 가상데이터의 분포도이다.

**[표 8] 정상 혈압과 고혈압 진단 기준**

| 혈압 | 수축기 혈압 | | 확장기 혈압 |
|---|---|---|---|
| 정상 | 120 미만 | AND | 80 미만 |
| 고혈압 전 단계 | 120 - 139 | OR | 80 - 89 |
| 고혈압 | 140 이상 | OR | 90 이상 |

[표 9] 20세 이상의 한국인 성인의 고혈압 유병률

| 성별 | 정상 | 고혈압 전 단계 | 고혈압 |
|---|---|---|---|
| | 분율 (표준오차) | 분율 (표준오차) | 분율 (표준오차) |
| 남자 | 35.2 (1.5) | 39.9 (1.5) | 24.9 (1.2) |
| 여자 | 59.2 (1.2) | 20.3 (0.9) | 20.5 (0.9) |
| 계 | 47.3 (1.1) | 30.0 (1.0) | 22.7 (0.8) |




[그림 17] 가상 데이터의 혈압 분포도

실험 데이터에 대한 클러스터링이 완료된 후 입력되는 혈압변화에 따라 어떤 보안 등급을 계산해 내는지 확인하였고, 또한 GRBAC 모델과 FCM 클러스터링 모델을 각각 적용했을 때의 혈압변화에 따른 보안등급 계산결과를 비교하였다. 표 10은 GRBAC 모델에 적용하기 위한 권한정책을 나타낸다. 보안등급은 총 5등급으로 분류되며 등급이 낮을수록 낮은 수준의 보안 서비스를 적용한다. 제안 모델에서 보안 등급은 사용자의 상황에 따라

결정되는데 사용자의 현재 상태가 나쁠수록 낮은 등급이 부여된다. 이는 긴급 상황의 경우, 보안 수준을 최소한으로 변경함으로써 사용자의 생명을 보호하기 위한 최대한의 정보를 제공하기 위함이다. 즉 사용자의 상황이 양호할 경우에는 개인 정보를 철저하게 보호할 수 있는 높은 수준의 보안 서비스를 제공하고, 긴급 상황의 경우에는 사용자의 보호를 최우선으로 고려함을 의미한다.

[표 10] GRBAC 모델에서의 상황 발생 시의 권한정책

| 환경정보 | | 작업 | 행동 주체 | 행동 대상 | 혈압 | 등급 |
|---|---|---|---|---|---|---|
| 시간 | 장소 | | | | | |
| 모든 시간 | 모든 장소 | 정보 조회 | 의료 요원 | 환자 정보 | 156 / 100 이상 ~ | 1 |
| | | | | | 151 / 98 ~ 155 / 99 | 2 |
| | | | | | 146 / 96 ~ 150 / 97 | 3 |
| | | | | | 141 / 94 ~ 145 / 95 | 4 |
| | | | | | ~ 140 / 93 | 5 |

## 2. 평가

표 11은 GRBAC 모델을 적용한 보안등급 계산결과와 FCM 클러스터링 알고리즘을 적용한 보안등급 계산결과를 나타낸다. 표 11에서 기대등급은 정상혈압수치와 비교하여 위험수치로 다가가는 정도에 따라 결정한 등급이다.

[표 11] 혈압변화에 따른 보안등급 계산 결과

| 혈압 | GRBAC | FCM 클러스터링 | 기대등급 |
|---|---|---|---|
| 152.5 / 98.8 | 2 | 3 | 3 |
| 155.6 / 97.8 | Unknown | 2 | 2 |
| 153.2 / 100.3 | Unknown | 3 | 3 |
| 151.8 / 99.8 | 2 | 3 | 3 |
| 167.7 / 109.5 | 1 | 1 | 1 |
| 157.3 / 105.3 | 1 | 2 | 2 |
| 120.8 / 82.5 | 5 | 4 | 5 |
| 141.5 / 87.4 | Unknown | 3 | 4 |
| 137.3 / 85.3 | 5 | 4 | 4 |

표 11에서 혈압수치 167.7 / 109.5의 경우, 위급 상황으로 인식하여 가장 낮은 수준인 1등급의 보안서비스가 적용됨을 알 수 있다. 또한 다소 높은 혈압범위에서는 2~3등급, 정상 혈압 범위에서는 4~5등급의 보안등급을 제시하고 있음을 확인할 수 있다.

그러나 GRBAC 모델을 적용한 보안등급 계산결과에서는 Unknown이 발생하고 있으며, 이는 환자의 혈압상태가 정규적으로 분류되지 않고 두 개 이상의 조건에 교차되어 있는 경우이다. GRBAC 모델의 권한정책을 더욱 상세하게 설정할 경우 적절한 보안등급의 도출이 가능하겠지만 모든 상황에 대하여 적절한 각각의 권한정책 설정은 현실적으로 매우 어려우며 또한 권한정책의 분류 개수가 많아질수록 결과 도출에 시간이 걸릴 것을 예측할 수 있다.

반면, FCM 클러스터링을 적용한 결과는 일반적으로 모든 상황에 대하여

적절한 결과를 보여주고 있다. 이는 자율적인 클러스터링을 수행한 후, 각 클러스터의 중심 값과 입력벡터의 거리를 이용하여 더욱 가까운 클러스터, 즉 더욱 적절한 등급을 제시하기 때문이며 고정된 규칙을 기반으로 한 GRBAC 모델보다 더욱 유연하고 적합한 결과를 보여준다.

또한 RFID/USN 환경에서는 센서 네트워크로부터 입력되는 다양한 종류의 상황정보를 기반으로 상황인식을 수행하게 되므로, 제안 모델에서도 다양한 종류의 상황정보가 각각 FCM 클러스터링을 이용하여 분류된다. 따라서 계산된 각각의 보안등급을 통합하여 보안 등급을 도출하여야 하며 제안 모델에서는 FCM 클러스터링을 통해 도출된 결과 데이터를 다시 퍼지 결정트리에 적용함으로써 입력된 모든 상황정보를 고려한 보안 등급을 도출한다.

고혈압 환자의 경우 가장 중요한 센서 정보는 혈압정보이지만 심박 수, 체온, 주변온도와 같은 정보의 영향을 많이 받게 된다. 또한 각 센서 정보별로 그 중요도가 다르기 때문에 FCM 클러스터링의 결과에 중요도를 표현하는 가중치를 곱한 값을 퍼지 결정트리의 입력 값으로 적용한다. 표 12는 혈압변화에 따른 보안등급과 심박 수 변화에 따른 보안등급을 퍼지 결정트리에 적용하여 도출된 최종 보안 등급의 계산 결과를 나타낸다.

[표 12] FCM 클러스터링과 퍼지 결정트리를 이용한 보안등급

| 혈압 | FCM등급 | 심박수 | FCM등급 | FDT등급 |
|---|---|---|---|---|
| 152.5 / 98.8 | 3 | 117 | 4 | 3 |
| 155.6 / 97.8 | 2 | 189 | 2 | 2 |
| 153.2 / 100.3 | 3 | 95 | 5 | 3 |
| 151.8 / 99.8 | 3 | 86 | 5 | 3 |
| 167.7 / 109.5 | 1 | 122 | 4 | 1 |
| 157.3 / 105.3 | 2 | 78 | 5 | 2 |
| 120.8 / 82.5 | 4 | 73 | 5 | 4 |
| 141.5 / 87.4 | 3 | 160 | 3 | 3 |
| 137.3 / 85.3 | 4 | 201 | 1 | 2 |

표 12에서 혈압수치 137.3 / 85.3의 경우, 혈압 변화에 따른 보안등급은 4등급이지만 심박 수의 변화에 따른 보안등급은 1등급으로 계산되어 최종 보안 등급은 2등급이 제시되고 있음을 확인할 수 있다. 이는 환자의 현재 혈압상태는 정상이나 높은 심박 수로 인하여 혈압상태가 악화될 가능성이 보안등급에 적용되었음을 나타낸다. 즉 제안 모델은 환자의 주위 상황을 인식함으로써 환자의 상태변화에 대한 파악과 예측을 통해 가장 적절한 보안등급을 제시하고 있음을 확인할 수 있다.

지금까지의 구현 및 실험 결과를 통해 제안 모델과 기존의 GRBAC 기반 상황인식 시스템과 비교하여 보면

첫째, FCM 클러스터링을 적용한 모델은 고정된 규칙을 기반으로 하지 않고 자율적인 클러스터링 방식을 이용하므로 예상하지 못한 상황에 대해서도 가장 적절한 분류결과를 도출할 수 있다.

둘째, 자율적인 분류 방식을 이용함에 따라 수많은 규칙을 직접 관리하지 않아도 적절한 보안 등급을 도출할 수 있으므로 관리상의 오버헤드 문제도 해결할 수 있다.

셋째, 퍼지 결정트리를 이용하여 다양한 상황 정보에 대한 분류 결과를 통합하여 최종적인 보안 등급을 도출할 수 있고, 또한 각 상황 정보 별 중요도를 적용함에 따라 더욱 현실적인 보안 등급의 적용이 가능하다.

결론적으로 FCM 클러스터링 알고리즘과 퍼지 결정트리를 이용한 상황인식 보안 서비스 시스템은 관리상의 오버헤드를 감소시켜주고 예상하지 못한 상황을 포함한 다양한 상황에 대해서 가장 현실적이고 적합한 보안 서비스를 제공해 준다. 표 13은 GRBAC 모델과 제안 모델의 비교내용을 보여준다.

[표 13] GRBAC 모델과 제안 모델의 비교

| 기준 | GRBAC 모델 | 제안모델 (FCM + FDT) |
|---|---|---|
| 자율성 | 사전 입력된 고정된 규칙을 이용하여 상황인식 | 자율적인 클러스터링과 퍼지 결정 트리를 이용하여 상황인식 |
| 대응능력 | 고정적인 규칙을 이용하므로 예상하지 못한 상황에 대한 대응이 불가능함 | 클러스터와의 거리를 이용하여 상황을 분류/인식 하므로 예상하지 못한 상황에 대해서도 대응이 가능 |
| 관리 편이성 | 규칙 및 정책을 직접 관리함 오버헤드 발생 가능성이 높음 | 규칙 및 정책을 직접 관리할 필요가 없음 |
| 통합성 | 상황정보의 다양성을 위해서 정책구조가 갈수록 복잡해 짐 | 다양한 종류의 상황정보를 통합하여 최적의 결과를 도출 |

# Ⅴ. 결론

본 논문에서는 사용자의 상황을 인식하고, 다양한 수준의 보안기술을 적용하기 위하여 FCM 클러스터링 알고리즘과 퍼지 결정트리를 이용한 상황인식 보안 서비스를 제안하였다. 제안 모델은 FCM 클러스터링 알고리즘을 통한 자율적인 상황의 분류와 다양한 센서 정보에 대한 분류 결과를 통합하는 퍼지 결정트리를 이용하여 가장 적절한 보안 등급을 제시한다. 그리고 지속적인 관리가 필요한 고혈압 환자에 대한 가상적인 주변상황을 구성하여, 제안된 모델이 얼마나 유연하고 적절하게 보안서비스를 적용할 수 있는가에 대한 실험 결과를 제공하였다.

기존에 연구되었던 GRBAC 모델과 비교할 때, FCM 클러스터링을 적용한 모델은 고정된 규칙을 기반으로 하지 않고 자율적인 클러스터링 방식을 이용하므로 예상하지 못한 상황에 대해서도 적절한 분류결과를 보여주고 있다. 또한 GRBAC 모델은 다양한 상황에 대응하기 위해서 계속적인 정책 및 규칙의 확장이 필요하므로 관리상의 오버헤드가 발생하는 문제점을 가지고 있지만, 제안 모델에서는 수많은 규칙을 직접 관리하지 않고 자율적으로 상황을 분류하므로 이러한 관리상의 오버헤드 문제도 해결할 수 있었다. 이는 제안 모델이 유연하면서도 자율적인 보안 서비스를 제공함과 동시에 관리에 대한 비용 효율적인 면에서도 좋은 결과를 제공한다는 것을 보여주고 있다.

또한 GRBAC 모델은 다양한 종류의 상황정보를 적용하기 위해서 상황정보의 종류가 늘어날수록 정책 및 규칙의 구조가 점점 복잡해지는 단점을 가지고 있으나 제안 모델은 퍼지 결정트리를 이용하여 다양한 상황정보를

통합하고 분석함으로써 최적의 이득을 얻을 수 있는 결과를 선택하여 제시하고 있으며 이러한 결과 도출에 의해 환자의 상태변화에 따라 예측 가능한 가장 적절한 보안등급을 제시하고 있음을 알 수 있다. 또한 다양한 상황정보에 대하여 각각 더욱 중요한 정보에 중요도를 적용함으로써 더욱 현실적인 보안 등급의 적용이 가능하다.

앞으로의 연구 방향은 사용자 별로 상이한 특징적인 데이터가 적용 가능한 시스템을 개발하기 위하여 제안 모델에 학습기능을 적용하여 보다 현실적이고 각각의 사용자 별로 차별화된 성능을 제공하는 시스템이 되도록 해야 할 것이다. 또한 보안만을 위한 시스템이 아닌 상황인식을 필요로 하는 다른 모든 시스템에도 적용 가능한 범용적인 시스템으로 발전시켜 나가야 할 것이다. 본 논문에서는 실제 환자의 의료정보 대신 통계결과를 기반으로 구성한 가상의 데이터를 이용하여 제안 모델을 분석하였으나, 향후에는 실제 데이터를 이용한 환경으로 확장하여 더욱 현실성 있는 연구를 수행하여야 할 것이다.

# 참고문헌


[1] 임신영, 허재두, 박광로, 김채규, "상황인식 컴퓨팅 기술 동향," [IITA] 정보통신연구진흥원 학술정보 - 주간기술동향 1142호.

[2] B. Schilit, N. Adams and R. Want, "Context-Aware Computing Applications," *In Proc of the 1st International Workshop on Mobile Computing Systems and Applications*, 1994, pp.85-90.

[3] A. K. Dey, Providing Architectural Support for Building Context-Aware Applications, PhD Thesis, College of Computing, Georgia Institute of Technology, 2000.

[4] J.A. Muhatadi, et al., "Cerberus: A Context-Aware Security Scheme for Smart Spaces," In Proc. of the First International Conference on Pervasive Computing and Communications (PerCom'03), 2002.

[5] M.J.Covington, et al., "A Context-Aware Security Architecture for Emerging Applications," In Proc. of the 18th Annual Computer Security Applications Conferences (ACSAC'02), 2002, pp. 249-258.

[6] 엄정호, 박선호, 정태명, "유비쿼터스 컴퓨팅 환경을 위한 보안통제가 강화된 접근제어 시스템 설계에 관한 연구," 정보보호학회논문지, 제18권, 제5호, 2008, pp.71-81.

[7] Kiyeal Lee, Seokhwan Yang, and Mokdong Chung "Context-Aware Security Service in RFID/USN Environments using MAUT and Extended GRBAC," *In Proc of the 2nd IEEE International Conference on Digital Information Management (ICDIM'07)*, Lyon, 2007, pp.303-308.

[8] R.L. Keeney and H. Raiffa, Decisions with Multiple Objectives:

Preferences and Value Tradeoffs (2nd Edition), Cambridge University Press, Cambridge, 1993.

[9] Mokdong Chung, Vasant Honavar, "A Negitiation Model in Agent-mediated Electronic Commerce," *In Proc of the IEEE International Symposium on Multimedia Software Engineering*, Taipei, Dec.2000, pp.403-410.

[10] 김환조, 채종우, 정목동, "X9.59를 이용한 DRM 기반 컨텐츠 유통 시스템 설계 및 구현," 한국멀티미디어학회 춘계학술대회 논문집, 제6권, 제1호, 2003, pp.771-774.

[11] 김환조, "적응적 보안 등급을 이용한 DRM 시스템 설계 및 구현," 석사학위논문, 부경대학교, 2004.

[12] 권중규, "RFID 애플리케이션을 위한 상황 인식 보안 아키텍처," 석사학위논문, 부경대학교, 2006.

[13] 남승좌, 박석, "유비쿼터스 컴퓨팅 환경의 역할기반 접근제어에서 발생하는 상황충돌," 정보보호학회논문지, 제15권, 제2호, 2005, pp.37-52.

[14] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E.Youman., "Role-based access control models," *IEEE Computer*, Vol. 29, No. 2, February 1996, pp. 38-47.

[15] http://csrc.nist.gov/rbac/NIST.

[16] M. J. Convington, et al., "Generalized Role-Based Access Control for Securing Future Applications," *In Proc of the 23th National Information Systems Security Conference(NISSC)*, Baltimore, 2000, pp.115-125.

[17] M. J. Moyer and M. Ahamad, "Generalized Role-Based Access

Control," *In Proc of IEEE International Conference on Distributed Computing Systems(ICDSC2001)*, 2001, pp.391-398.

[18] M. J. Convington, et al., "A Context-Aware Security Architecture for Emerging Applications," *In Proc of the 18th Annual Computer Security Applications Conferences(ACSAC'02)*, 2002, pp.249-258.

[19] 임현수, 조은애, 문창주, "RBAC에서 권한 할당 제약사항들 간의 충돌 탐지 모델," 한국시뮬레이션학회 2005년 추계학술대회 논문집, 2005, pp.51-55.

[20] J. Bezdek, "A convergence theorem for the fuzzy ISODATA clustering algorithm," *IEEE Trans. Pattern Anal. Machine Intelligence*, Vol.PAMI-2, No.1, 1980, pp.1-8.

[21] 오성권, 프로그래밍에 의한 컴퓨터지능, 내하출판사, 2002.

[22] 이우향, 이건명, "특징공간을 사선 분할하는 퍼지 결정트리 유도," 정보과학회논문지 : 소프트웨어 및 응용, 제29권, 제3호, 2002, pp.156-166.

[23] 이건명, "퍼지 데이타에 대한 퍼지 결정트리 기반 분류규칙 마이닝," 정보과학회논문지 : 소프트웨어 및 응용, 제28권, 제1호, 2001, pp.64-72.

[24] 이현숙, "점증적 학습 퍼지 신경망을 이용한 적응 분류 모델," 퍼지 및 지능시스템학회 논문지, 2006, 제16권, 제6호, pp.736-741.

[25] 전문진, 도준형, 이상완, 박광현, 변증남, "다변량 퍼지 의사결정트리와 사용자 적응을 이용한 손동작 인식," 로봇공학회 논문지, 제3권, 제2호, 2008, pp.81-90.

[26] David Hook, Beginning Cryptography with Java, Wiley, 2005.

[27] Jess Garms, Daniel Somerfield, Professional Java Security, Wiley,

2002.

[28] 양석환, 이기열, 정목동, "RFID/USN 미들웨어에서의 응용S/W 보안," 한국컴퓨터종합학술대회논문집, 제34권, 제1호, 2007, pp.84-88.

[29] E. Brickell, J. Camenisch, and L. Chen, "Direct anonymous attestation," *In Proc of 11th ACM Conference on Computer and Communications Security*, ACM Press, 2004 Practical Solutions to Identification and Signature Problems, ACPC 86, LNCS, 1987.

[30] TCG, TCG Specification Architecture Overview Specification Revision 1.3, 2007.

[31] Ronald Cramer, "Signature Schemes based on the Strong RSA Assumption," ACM TISSEC 3(3):161-185, 2000; extended abstract *in Proc 6th ACM Conference on Computer and Communications Security*, 1999.

[32] 양석환, 이기열, 정목동, "DAA 프로토콜을 위한 실험 모듈 설계 및 구현," 한국정보과학회 학술심포지움 논문집, 제1권, 제1호, 2007, pp.42-46.

[33] "국민건강영양조사 제3기(2005) 검진조사," 보건복지부 질병관리본부, June 2006, pp.114-115.

# 감사의 글

먼저 2년간의 대학원 생활을 무사히 마칠 수 있게 도와주신 모든 분들께 감사의 말씀을 드립니다. 7년간의 공백 후에 학교로 돌아온 저에게 대학원은 낯설면서도 반드시 이겨내야만 하는 어려운 장소였습니다. 또한 학교라는 특성으로 인해 자칫 해이해 질 수도 있는 2년을 많은 분들의 도움으로 인해 무사히 마칠 수 있었습니다.

특히 언제나 깊은 관심을 가지고 지도를 해 주신 정목동 교수님께 진심으로 감사를 드립니다. 단지 학업만이 아니라 제가 고민했던 많은 부분에서 교수님의 말씀은 제가 올바른 길을 찾을 수 있는 길잡이가 되었습니다. 그리고 바쁘신 와중에도 부족한 저의 논문을 심사해 주신 신봉기 교수님, 송하주 교수님께도 마음 깊이 감사드립니다.

또한 함께 프로젝트를 수행하며 많은 도움을 주었던 이현동, 안규희, 이기열, 김건아, 서대교, 신승목, 강호식, 우준혁, 오송이, 양귀열, 여현우, 이동훈, 이승철 등 연구실 멤버들과 선배들께도 고맙다는 말을 하고 싶습니다.

마지막으로 회사생활을 접고 다시 학교로 돌아가 버린 저를 끝까지 믿고 지지해 주신 아버지와 부족한 아들의 가는 길을 하늘에서 따뜻하게 지켜보고 계실 어머니, 그리고 누나, 동생 홍준에게 감사의 마음과 함께 사랑한다는 말을 전하고 싶습니다.

2008년 12월

양석환 드림