Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.25, NO.2, Apr. 2015 ISSN 2288-2715(Online)
http://dx.doi.org/10.13089/JKIISC.2015.25.2.439

IoT에서 Capability 토큰 기반 접근제어 시스템 설계 및 구현*

이 범 기,† 김 미 선, 서 재 현‡
국립목포대학교

Design and Implementation of The Capability Token based Access Control System in the Internet of Things*

Bum-Ki Lee,† Mi-Sun Kim, Jae-Hyun Seo‡
Mokpo National University

요 약

사물인터넷은 모든 고유 식별 가능한 임베디드 컴퓨팅 장치가 기존의 인터넷 환경에 연결된 초연결 사회를 지향한다. 따라서 사물인터넷 서비스는 단순한 사물간의 통신(M2M)을 넘어 많은 프로토콜들, 도메인들, 애플리케이션들 위에서 유연한 통신 능력을 가지고 사용자에게 권한을 부여할 수 있어야한다. 또한, 사물인터넷의 접근제어는 보안과 신뢰성을 증가하기 위하여 전통적인 접근제어와 차별화된 방법이 필요하다. 본 논문에서는 사물인터넷 환경에서 안전한 접근제어를 위하여 capability 토큰 기반의 시스템을 설계하고 구현하였다. 제안한 시스템에서 capability 토큰은 인가권한을 의미하며, 토큰은 위임, 재위임, 폐기 가능하다. 제안된 시스템은 capability 토큰을 사용함으로써 접근 제어 처리 시간을 감소할 수 있을 것으로 기대한다.

ABSTRACT

IoT (Internet of Things) propels current networked communities into a advanced hyper-connected society/world where uniquely identifiable embedded computing devices are associated with the existing internet infrastructure. Therefore, the IoT services go beyond mere M2M (Machine-to-Machine communications) and should be able to empower users with more flexible communication capabilities over protocols, domains, and applications. In addition, The access control in IoT need a differentiated methods from the traditional access control to increase a security and dependability. In this paper, we describe implementation and design of the capability token based system for secure access control in IoT environments. In the proposed system, Authorities are symbolized into concepts of the capability tokens, and the access control systems manage the tokens, creation, (re)delegation and revocation. The proposed system is expected to decrease the process time of access control by using capability tokens.

Keywords: Internet of Things, Access Control, Capability Token, ACL, CL

접수일(2015년 2월 11일), 수정일(2015년 4월 2일), 게재확정일(2015년 4월 2일)
* 본 논문은 2014년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. NRF-2014R1A2A1A11053774)
† 주저자, leebumki@mokpo.ac.kr
‡ 교신저자, jhseo@mokpo.ac.kr(Corresponding author)

I. 서 론

사물인터넷(Inertnet of Things : IoT)은 정보통신기술 발전에 따라 모든 사물과 사람이 네트워크로 연결되는 초연결 사회를 지향한다. 사물인터넷은 통신모듈, 센서, 단말기를 통해 서비스가 가능하며,

교통카드, 바코드, 물류추적, 가로등 원격 제어 및 공장 설비와 같은 산업현장에서 활용되고 있다.

그러나 사물인터넷은 많은 기술요소들이 접목되고 많은 정보들이 발생, 교환됨으로써 보안위협 요소도 상대적으로 증가할 것으로 예상된다. 사물인터넷의 여러 보안 이슈 중에서 본 논문에서는 사물인터넷 환경에서 접근제어 기술에 대해서 연구하였다.

본 논문에서는 짧은 시간에 다양한 연산이 이루어지는 사물인터넷 환경에 적합한 접근제어 시스템을 설계하고 구현한다. Capability 토큰을 활용한 접근제어를 통해 접근제어의 경량화 프로세스가 가능하도록 설계하였다. 사물인터넷 환경과 같은 접근 요청이 빈번히 발생할 수 있는 환경에서 토큰을 통해 객체에 접근을 허용함으로써 접근제어 처리 시간을 감소시킬 수 있다. 생성된 capability 토큰은 다른 사용자에게 위임, 재위임 될 수 있으며, 사용자에 의해 폐기 가능하다. Capability 토큰의 관리를 위해 위임가능 여부 및 유효기간 등의 정책이 설정될 수 있으며, 폐기에 대한 정책도 설정할 수 있다.

Capability 접근제어는 짧은 시간에 다양한 연산이 이루어지는 사물인터넷 환경에서 장치들의 효율적인 관리를 제공할 수 있을 것이다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 사물인터넷 환경의 접근제어 이슈 및 기존 접근제어 기술에 대해 설명한다. 3장에서는 capability 토큰을 사용하는 접근제어 시스템 설계를 제시하고, 4장에서는 제안한 시스템에 대한 구현 및 평가결과를 보여준다. 마지막으로 5장에서는 결론과 미래 연구방향에 대해 제시한다.

II. 관련연구

2.1 사물인터넷 환경의 접근제어

사물인터넷 환경에서는 사람, 사물, 데이터와 같은 모든 객체가 인터넷과 연결되기 때문에 각 객체간의 프라이버시 및 보안의 문제가 중요한 영향을 미친다. 다라서, 공격, 데이터 인증, 접근제어 및 프라이버시에 대한 확실한 방법이 설립될 필요가 있다[1].

사물인터넷 환경에서 공격, 데이터 인증, 접근제어 및 프라이버시에 대한 방법을 설립하기 위해서는 기존 인터넷 환경과는 다른 변화를 감지하고 예상하여 고려하여야 한다.

첫째, 사물인터넷은 기존 인터넷 환경과 달리 짧은 시간동안 상호 작용이 일어나며, 동일한 요청이 자주, 자발적으로 수행될 수 있다.

둘째, 사물인터넷에서 자원/서비스/오퍼레이션/데이터 등에 대한 분석 및 인가는 같은 요청에 대해서도 고정적이지 않고, 주변의 상황에 따라 바뀔 수 있다.

따라서 사물인터넷과 같이 개방되고, 광범위한 컴퓨팅 환경에서는 확장성의 문제, 장치들의 관리의 문제, 유연성 있고 쉬운 권한 위임의 문제를 고려한 접근 제어 기법이 필요하다[2].

적절한 권한 위임, 권한 관리를 통해 객체에 대한 불필요한 접근을 제한하고, 사물인터넷 환경에서 발생하는 방대한 범위의 데이터에 대해 위임된 권한을 통한 접근 관리가 필요하다.

2.2 접근제어 메커니즘

접근제어는 주체가 정책에 따라 객체의 작업을 수행 할 수 있는지 여부를 나타내는 것으로, 자원에 대한 인가되지 않은 접근을 감시한다. 접근 요청에 대한 이용자를 식별하며, 접근요청이 정당한 것인지를 확인하여 기록하고 보안정책(security policy)에 따라 접근 승인 또는 거부함으로써 비인가자로부터의 불법적인 자원접근 및 파괴를 예방한다. 즉 접근통제는 각 자원에 대한 기밀성, 무결성, 가용성 및 합법적인 이용과 같은 정보보호 서비스에 권한부여를 위한 수단이 된다[3][4]. 접근제어는 접근제어 정책, 접근제어 모델, 접근제어 메커니즘으로 정의하여 분류한다. 접근제어 메커니즘은 시도된 접근 요청을 정의된 규칙에 대응시켜 검사함으로써 불법적 접근을 방어하는 것이다. 접근제어 메커니즘은 Access Control List(ACL), Capability List(CL), Security Label(SL)로 구분한다[5][6].

2.2.1 Access Control List(ACL) 기반 접근제어

ACL(접근 제어 리스트)은 객체나 객체 속성에 적용되어 있는 허가 목록을 말한다. 이 목록은 누가 또는 무엇이 객체 접근 허가를 받는지, 어떠한 작업이 객체에 수행되도록 허가를 받을지를 지정하고 있다[7].

Fig. 1.은 ACL을 이용한 접근제어의 예를 보여주고 있다. 접근의 대상이 되는 리소스 A, 즉 객체가가 권한 목록을 가지고 있으며, 주체가 접근 요청

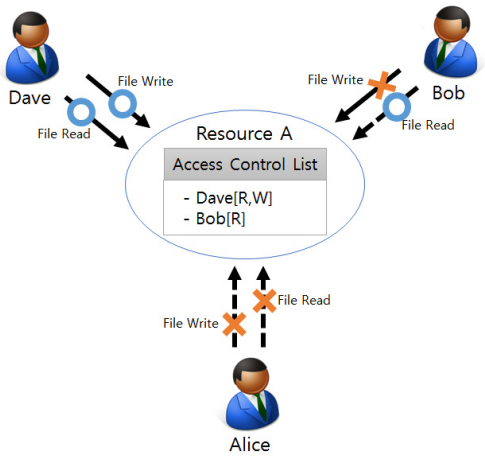

Fig. 1. Example of The access control used ACL

을 할 경우, 객체가 가지고 있는 권한 목록을 확인하여 주체의 요청에 대해 허가/거부를 수행한다.

ACL은 주체, 객체의 수가 매우 많은 시스템에서는 행렬 또는 목록의 수가 많아짐에 따라 탐색에 많은 시간이 소요되는 단점을 갖는다.

2.2.2 Capability List(CL) 기반 접근제어

CL 기반 접근제어는 객체에 대한 접근 권한을 정의한 Capability를 주체가 갖는 방식으로 기존 ACL 접근제어 방식과 다르다. Capability는 권한 정보 집합을 의미하며, CL 기반 접근제어의 특징은 최소 권한의 원칙과 높은 유용성 및 유연성의 정도에 따른 관련 기능을 제공하여 미래 인터넷의 요구 사항에 적합하다.

CL은 Table 1.에서 제시한바와 같이 접근제어행렬(ACM)에서 각 행에 해당한다. Capability는 주체가 새로운 객체를 생성할 수 있고, 또한 그 객체에게 허용되는 권한을 정의할 수 있다[6].

Table 1. Example of Capability List

	Resource 1	Resource 2	Resource 3
User A	Read, Write	Read, Write	Read, Write
User B	-	Read, Write	Read, Write
User C	-	-	Read, Write

2.2.3 ACL과 CL 기반 접근제어 비교

Fig. 2.는 ACL과 CL 기반 접근제어를 비교하여 도식화 한 것으로, ACL 접근제어에서는 사용자가 자원에 대한 연산을 요청할 경우에 서비스 제공자는 사용자가 직접 또는 간접적으로 객체에 대한 권한이 있는지의 여부를 확인한 후, 요청한 자원 또는 요청한 연산을 수행할 수 있도록 인가한다. 이와 달리 CL 기반 접근제어에서는 사용자가 자신이 갖고 있는 capability를 서비스 제공자에게 제시하면 서비스 제공자는 capability를 확인하여 요청한 자원 또는 요청한 연산을 수행할 수 있도록 인가한다[8][9].

권한에 대한 검증은 ACL 접근제어의 경우 접근 대상이 되는 객체가 갖는 ACL을 통해 객체가 검증하게 된다. 반면, CL 기반 접근제어의 경우 각 객체에 접근하고자 하는 주체가 CL을 갖으며 Capability를 제시함으로써 각 객체에 접근한다.

ACL과 CL 기반 접근제어는 앞에서 살펴본바와 같이 인가하는 과정에서도 차이가 나타나지만, 권한을 부여하는 방식도 서로 차이를 보인다.

Fig. 3.은 ACL 기반 권한 부여에 대한 구조도이다. 전통적인 ACL 모델에서 권한은 자원과 동일한 관리 도메인 부분이라고 할 수 있다. 자원 제공자는

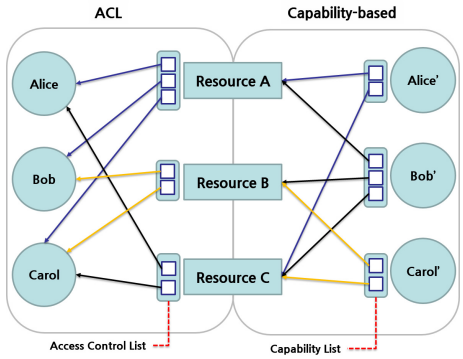

Fig. 2. Compare ACL with CL based access control

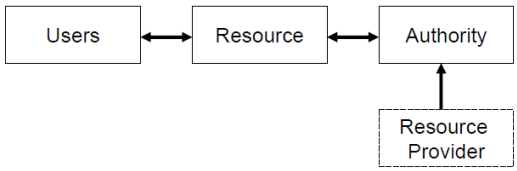

Fig. 3. ACL-based Authorization Infrastructure

권한에 대한 관리 권한이 없다. 따라서 권한 부여 정책을 설정하기 위해서는 자원 제공자와 사용자 관리자의 승인 과정이 필요하기 때문에 자원 제공자는 권한을 통해 자원의 접근여부를 결정한다. 반복되는 작업의 경우 이 단계는 지속적으로 발생할 수 있다.

Fig. 4.는 CL 기반 접근제어를 단순화하여 나타내었다. 워크플로우(workflow)는 다음과 같다.

사용자는 먼저 자원제공자(인증기관)에 접근 요청을 보낸다. 자원제공자는 접근 요청을 허가하는 capability 또는 접근 요청 허가에 대한 capability를 발행할 수 있는 권한을 위임할 수 있다. 이 후, 사용자는 자신의 capability와 함께 자원에 대한 접근 요청을 보낸다. 자원에서 capability의 검증을 통해 자원에 대한 액세스 여부를 결정한다.

Capability는 재사용이 가능하다는 이점을 갖는다. 작업의 반복이 발생할 경우 ACL 기반 시스템에서는 반복적으로 인증 프로세스가 진행되지만, CL 기반 접근제어 시스템에서는 기발행된 capability를 통해 반복 작업을 최소화 할 수 있다. 따라서 ACL 기반 시스템에 비해 워크플로우가 가볍다고 할 수 있다[10].

Fig. 4. CL-based Authorization Infrastructure

III. Capability 토큰 기반 접근제어 시스템 설계

사물인터넷 환경에서 접근제어는 다양한 단말간 통신이 발생하고, 더 많은 객체간의 통신이 이루어지기 때문에 전통적인 접근제어와 차별화된 방법이 필요하다.

CL 기반 접근제어는 주체에게 부여된 권한을 통해 주체가 객체에 접근하며 이러한 권한은 다른 주체에 위임이 가능하다.

본 논문에서는 CL 기반 접근제어를 사물인터넷 환경에서 적용하고자 하며, 주체의 권한에 대해 capability 토큰을 생성하고, 이를 이용하여 접근제어를 수행하는 시스템을 설계하였다.

제안된 시스템은 사물인터넷 내에서 주체의 권한을 capability 토큰으로 정의하며, 다양한 주체들이 생성된 capability 토큰을 객체에 전달하여 접근 여부를 결정한다. 주체와 객체가 많아지는 사물인터넷 환경에서 접근 요청이 있을 때마다, 객체가 주체에 대한 권한을 확인한 후 접근 여부를 결정하는 기존 접근 제어 방식과 달리 제안된 시스템에서는 사용자는 capability 토큰을 제시하고 이에 대한 유효성 검증만으로 객체에 대한 접근 여부가 결정되므로, 접근제어 프로세스의 경량화를 제공한다.

사물인터넷 환경과 같이 접근 요청이 빈번히 발생할 수 있는 환경에서 capability 토큰을 전송하고 capability 토큰의 무결성만을 보장 받아 객체에 접근을 허용함으로써 접근제어 처리 시간을 감소시킬 수 있다.

또한, capability 토큰은 위임, 재위임, 폐기를 통해 관리상의 효율성을 제공한다. Capability 토큰에 부여된 권한을 다른 장치에 위임함으로 위임 받은 장치는 위임된 권한의 전부 또는 일부를 활용할 수 있다.

Fig. 5.는 각 모듈을 통합하여 나타낸 구성도이다. 단말에서 접근제어 토큰을 부여하는 과정을 통합적으로 나타내고 있다.

제안한 capability 토큰 기반 접근제어 시스템은 사용자 인증 및 권한 부여를 위해 정책집행점(PEP), 인가 엔진(Authorization Engine), 인증 모듈(Authentication Module) 및 접근제어 모듈(Access Control Module)로 구성된다.

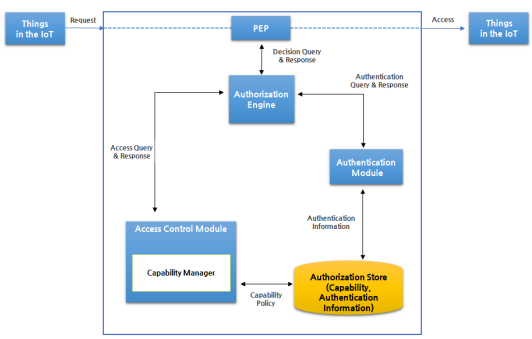

Fig. 5. Proposed System Architecture

3.1 인가 엔진

인가엔진은 접근제어 모듈 및 인증 모듈과 직접적으로 통신하며 결정된 인가 결과를 정책집행점에 전달하는 기능을 수행한다.

Fig. 6.은 정책집행점(PEP)와 인가엔진 사이의 관계를 보이고 있으며, PEP는 단말(Things)의 접

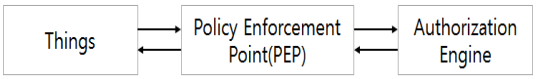

Fig. 6. Policy Enforcement Point(PEP) Infrastructure

근제어 요청 시 사용자 인증 및 접근제어 토큰에 대한 요청을 인가엔진에 전달한다. 인가엔진은 하위 모듈과의 통신을 통해 요청에 대한 응답을 정책집행점에 전달하고, PEP는 이를 이용하여 단말의 요청을 집행한다.

Fig. 7.과 같이 인가엔진에서는 PEP로부터 받은 단말에서 요청한 정보를 통해 인증 또는 접근제어를 확인하기 위한 처리를 수행한다. 인가엔진은 인증모듈 및 접근제어 모듈과 직접적으로 통신하여 단말의 접근 요청을 수행한다.

또한, 인가엔진은 인증모듈과 접근제어모듈의 중간에서 두 모듈 사이의 통신을 중재한다. 인증모듈에서의 결과 값이 인증된 사용자일 경우 접근제어 모듈에 이를 전달하여 해당 사용자의 권한 토큰을 발급되도록 하거나, 토큰을 이미 갖고 있는 단말일 경우는 해당 토큰에 대한 유효성 검사를 수행할 수 있도록 중재한다.

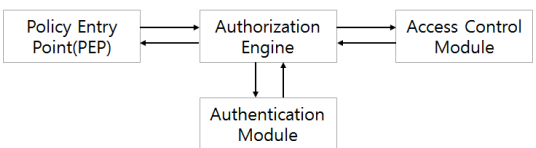

Fig. 7. Authorization Engine Infrastructure

3.2 인증 모듈

사용자 권한의 식별을 위해서는 먼저 접근 가능한 사용자인지에 대한 확인이 필요하다. 인증 모듈은 최초의 사용자 입력정보를 통해 접근가능 여부를 식별하여 사용자에게 응답하며 Fig. 8.과 같다.

인가엔진에서 인증모듈로 단말의 인증정보를 이용하여 사용자 식별을 요청할 경우, 인증모듈의 컨트롤러를 통해 인증정보를 받아 인증처리기를 통해 인가스토어로 등록여부 확인한다. 검증 결과를 인가 응답기를 통해 컨트롤러로 반환하고 컨트롤러는 인가 응답에 대한 결과를 인가 엔진으로 전달하여 사용자 인증 수행한다.

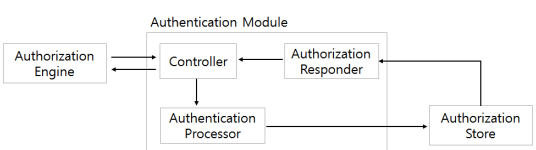

Fig. 8. Authentication module Infrastructure

3.3 접근제어 모듈

접근제어모듈은 인가엔진으로부터 인가된 사용자 정보를 식별하여 토큰을 발행하고 검증하는 역할을 수행한다. 접근제어 모듈의 구조도는 Fig. 9.와 같으며, 인가 엔진으로부터 받은 사용자 인증 정보는 접근제어 모듈의 컨트롤러를 통해 capability 관리자에 전달되고, capability 관리자는 토큰 생성, 발행 및 위임 등의 토큰 관리를 수행한다.

관리대상 토큰은 capability 관리자에서 토큰 처리기에 전달하여 인가 스토어에 해당 토큰 정보를 갱신하여 토큰정보를 관리한다. 생성된 토큰 및 토큰 검증 정보는 컨트롤러를 통해 인가 엔진에 전달된다.

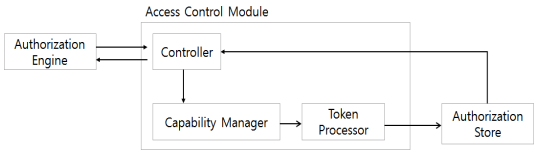

Fig. 9. Access control module Infrastructure

3.3.1 토큰 발행

본 논문에서 제안한 capability 토큰은 디지털 인증서 표준인 X.509와 유사하게 설계하였다. 토큰의 구조 중 차이점은 위임에 대한 깊이를 제한하고 있는 것으로 깊이에 따라 권한위임을 제한 할 수 있다.

토큰 정보를 통해 접근제어 및 위임은 각 모듈을 통해 검증되고 인증된다. 일련의 과정은 capability 토큰 기반 접근제어를 통해 권한 위임, 폐기 및 세분화된 권한 할당이 가능하다. 이를 통해 불필요한 접근제어를 최소화가 가능하며, 권한 위임에 따라 소멸되어야 할 데이터에 대한 관리가 가능하다.

토큰 발행은 인증된 단말 중 토큰이 없을 경우 발생하는 이벤트로, 인가 스토어의 정보를 통해 권한을 확인하고 토큰을 생성한다. Capability 토큰의 구조는 Table 2.와 같다.

토큰은 capability 관리자에 의해 발행 및 관리

Table 2. Capability token structure

Filed	Description
SerialNumber	Unique serial number of the certificate
Issuer	The Issuer of certification
Validity	The validity period of the certificate
Subject	Information of Subject
Object	Target object
Access Right	Right for Accessible objects
Owner	Information of Resource Owner
Right Depth	The current level of delegated authority
Limit Depth	The maximum possible level of authority delegated
HashSignature	Token Hash Signature

되며, capability 관리자에서 발급한 토큰을 인가엔진으로 전달한다. 단말은 발급된 토큰을 통해 접근권한을 얻는다.

토큰의 발행 과정은 Fig. 10.과 같으며, Capability Manager의 경우 접근제어 모듈 내에 포함되어 있으나, 시스템의 흐름을 명확하게 보이기 위하여 명시하였다.

① 인가엔진에서 접근제어모듈에 사용자의 ID 또는 단말의 장치식별자를 전달한다.
② 접근 제어 모듈 내에서 capability 관리자에 인증 정보를 전달하며, 토큰의 등록 여부를 확인 요청한다.
③ Capability 관리자는 인가 스토어에 사용자 정보에 대한 토큰 식별 및 결과를 수신한다.
④ Capability 관리자는 토큰 정보가 없는 경우 capability 토큰을 생성하여 접근 제어 모듈로 전달한다.
⑤ 접근 제어 모듈에서는 인가엔진으로 토큰 전달 및 접근제어에 대한 응답을 수행한다.

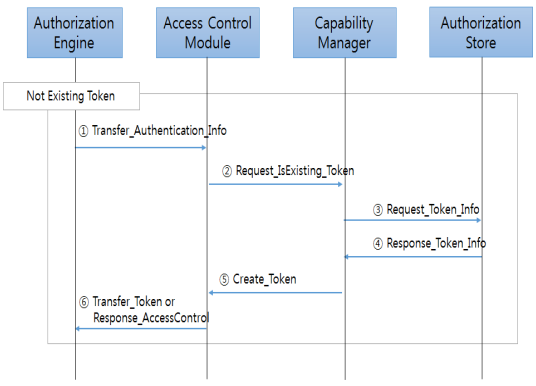

Fig. 10. Sequence Diagram of Token Creation

3.3.2 토큰 검증

발행된 토큰이 존재하는 경우, 토큰에 대한 검증을 수행하게 되며, Fig. 11.은 토큰 검증 순서를 나타내며, 토큰의 검증 순서는 다음과 같다.

주체가 권한에 대한 토큰을 소유하고 있을 경우, 객체에 대한 접근 시 토큰 유효성 및 권한검증은 인가스토어를 통해 식별한다. 접근제어 모듈에서 capability 토큰의 유효성 검증 프로세스 과정은 다음과 같다.

① 접근요청 발생 시 토큰의 등록여부를 인가스토어를 통해 확인한다.
② 토큰이 등록되어 있을 경우 토큰에 대한 정보를 로드한다.
③ 로드된 정보에서 토큰에 포함하고 있는 접근하고자 하는 객체가 맞는지 확인한다.
④ 접근하고자 하는 객체가 일치한다면, 해당 객체에 대한 토큰의 유효성을 검증한다.
⑤ 토큰 유효성이 확인되면, 접근권한에 따라 접근을 허용한다.
⑥ 확인 또는 유효하지 않는 토큰의 경우 접근은 거부된다.

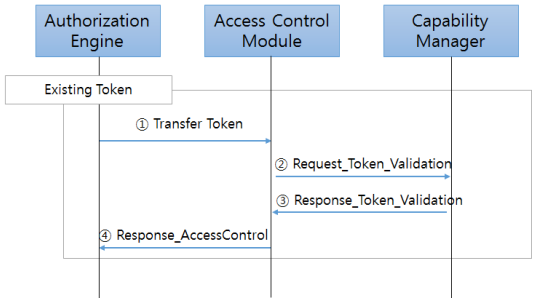

Fig. 11. Sequence Diagram of Token Validation

3.3.3 권한 위임 및 폐기

Capability 토큰 기반 접근제어 시스템의 특징 중 하나는 권한 위임 및 폐기가 가능하다는 것이다.

본 시스템에서 사용자의 접근 권한은 capability

토큰에 의해 결정되며, 자원 제공자는 capability 토큰의 생성뿐만 아니라 capability 토큰의 위임 및 폐기여부를 결정할 수 있다. Capability 토큰에 대한 위임 가능 여부는 capability 토큰 생성 시 결정되며, capability 토큰을 부여받은 주체는 위임 가능 여부에 따라 다른 주체에게 위임할 수 있다. 또한, 위임된 capability 토큰은 자원 제공자 및 권한 위임자에 의해 폐기될 수 있다.

Fig. 12.는 토큰의 위임 또는 폐기 요청 시 발생하는 과정을 보이고 있으며, 이에 대한 처리과정은 다음과 같다.

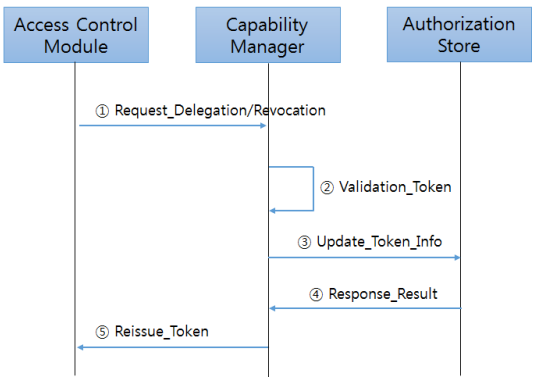

Fig. 12. Sequence Diagram of Token Revocation

IV. Capability 토큰을 활용한 접근 제어 시스템 구현 및 평가

4.1 Capability 토큰 기반 접근제어를 적용한 시스템 구현

본 논문에서는 제안된 capability 토큰 기반 접근 제어 시스템을 구현하기 위하여, Fig. 13.과 같이 라즈베리파이와 LED센서로 구성된 간단한 테스트베드를 구축하였다.

구축된 시스템은 웹을 통해 라즈베리파이의 GPIO를 제어 할 수 있는 WebIOPi를 사용하였으며, 주체는 capability 토큰에 의해 LED 센서에 접근할 수 있고, capability 토큰은 웹 페이지를 통해 생성, 위임, 폐기된다.

Capability 토큰의 위임가능 여부에 따라서 권한 위임이 가능하고, 각 권한에 따라 상태 확인(읽기) 및 제어(쓰기) 권한이 부여한다.

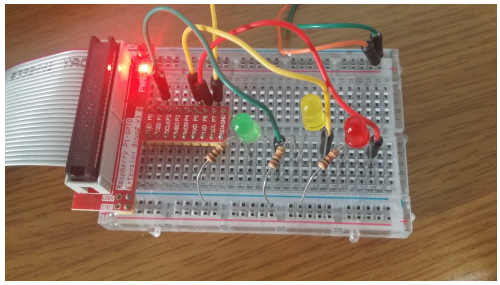

Fig. 13. Testbed for Proposed System

객체는 LED 모듈로 가정하여 구성하고 적색, 녹색, 황색의 객체로 구분한다.

4.1.1 토큰 등록

Fig. 14.는 인가스토어에 토큰을 등록하기 위한 관리 페이지이다. 토큰을 등록할 수 있는 권한의 관리자가 토큰에 필요한 정보를 입력하여 인가스토어에 저장한다. 일련번호(serial number) 및 토큰의 서명 값은 자동으로 생성하여 저장된다.

일련번호는 16자리, 토큰의 서명 값은 64자리로 SHA-256 해시 알고리즘을 활용하였다. 토큰 등록을 위한 정보 입력 후 인가스토어에 저장하기 위한

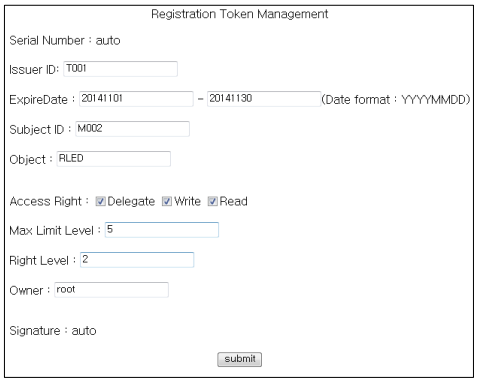

Fig. 14. Management page for token registration

Fig. 15. Registration result of token

토큰 정보 확인하면 Fig. 15.와 같이 등록 결과를 확인 할 수 있다.

4.1.2 토큰 정보 및 검증

토큰정보는 Fig. 16.과 같은 형태로 나타난다.

XML 형태로 토큰의 구조를 식별 할 수 있도록 하였다. 주체에게 발급되는 정보로 객체에 대한 권한의 수가 증가 할 경우 권한정보인 〈Capability〉〈/Capability〉 부분도 증가하게 된다.

토큰 검증을 위해 인가스토어에 저장된 서명 값과 발급받은 토큰의 서명 값을 비교하여 토큰에 대한 최종적인 유효성을 검증한다. 각 항목에 대한 값의 위변조 발생 시 서명 값은 달라지기 때문에 서명 값이 변경된다. 따라서 이를 통해 토큰의 신뢰성을 확보할 수 있다.

Fig. 16. Token Infrastructure

4.1.3 권한 위임

Fig. 17.은 토큰의 권한을 위임하기 위한 관리 페이지이다. 위임 권한이 있는 현재 사용자의 ID가 Issuer ID로 등록되며, 위임하고자 하는 객체가 Object에 라디오버튼 형태로 선택되어 나타난다. 접근 권한은 위임자의 권한보다 같거나 낮은 수준에서 위임자가 임의로 선택할 수 있다.

4.1.4 권한 폐기

인가된 사용자가 위임한 권한에 대해 폐기 할 수 있다. 사용자가 권한 폐기 관리페이지에 접속할 경우 Fig. 18.과 같이 인가스토어에 저장된 위임한 권한 목록이 나타나며 해당 권한을 폐기 할 수 있다.

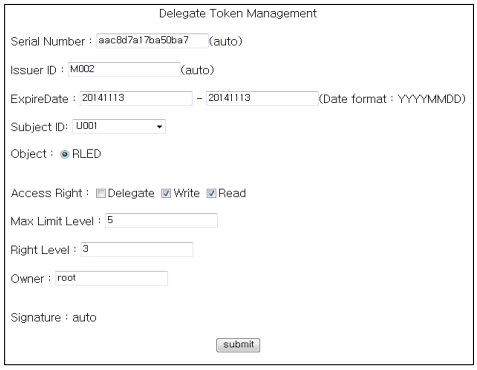

Fig. 17. A management page for token authority delegation

Fig. 18. A management page before token authority revocation

Fig. 19. A management page after token authority revocation

권한 폐기 시 인가스토어에서는 사용여부에 따른 상태 값이 기존에는 활성화되어 있다가 권한 폐기가 발생하면 비활성화 되도록 값을 수정한다.

권한 폐기 이후 폐기된 토큰은 사용할 수 없게 되며, 위임한 사용자가 권한 폐기를 위한 관리페이지 접속 시 해당 토큰이 나타나지 않는다. Fig. 19.는

권한 폐기 후 변경된 권한 폐기가 가능한 목록을 나타낸다.

4.2 시스템 분석 및 평가

4.1.1 ACL과 제안 접근제어 비교 분석

Table 3.은 ACL과 제안한 Capability 토큰 기반 접근제어를 비교하였다. 기존의 접근제어 방식에서 정보권한은 자원 제공자에게 있었으나, Capability 토큰 기반 접근제어에서는 사용자가 해당 권한을 갖는다. ACL에서 권한폐기는 관리자가 권한을 제거한다는 관점에서 가능하다고 할 수 있으며 제안한 접근제어에서는 토큰을 발행한 발행자에 의해 권한 폐기가 가능하다. ACL에서는 권한위임, 권한 재위임 및 위임권한에 대한 폐기는 불가능하였으나, 제안한 접근제어는 권한위임, 권한재위임 및 권한폐기가 가능하다.

제안한 접근제어 시스템은 권한이 객체에게 있는 기존의 접근제어와 달리 주체에게 있기 때문에 객체 수가 많은 사물인터넷 환경에서 적합하다. 또한 권한의 위임뿐만 아니라 재위임 및 권한 폐기가 가능하다는 장점이 있다.

사물인터넷 환경은 사물 간 인터넷 결합의 증가로 인해 유연한 접근제어 및 권한 관리가 필수적이다. 이로 인한 데이터 폭발은 다양한 데이터의 수집에 따라 개인정보 등과 같은 정보 관리에 있어서도 문제를 야기할 수 있다. 따라서 CL 토큰 기반 접근제어를 통해 유연한 권한 관리 및 접근제어를 통해 데이터에 대한 접근을 관리한다.

Table 3. Comparison of ACL and Capability Token

Property	ACL	Capability Token
Right's subject	resource provider	user
Rights revocation	yes	yes
Rights delegation	no	yes
Rights Re-Delegation	no	yes
Rights Delegation-Revoke	no	yes

V. 결 론

본 논문에서는 사물인터넷 및 기존 접근제어 방법에 대해 연구하고, Capability 토큰 기반 접근제어 시스템을 설계하였다. 그리고 구현한 접근제어 시스템을 사물인터넷 환경에서 적용 가능하도록 테스트베드를 구축하여 구현하였다.

기존 접근제어인 ACL은 권한에 대한 위임, 재위임 및 폐기 등이 불가능 하였다. 따라서 다양한 주체와 객체가 존재하는 사물인터넷 환경에서 기존의 접근제어 방법은 한계가 있다. 하지만 Capability 토큰 기반 접근제어 시스템을 통해 토큰은 다른 주체에게 위임, 재위임 될 수 있으며, 권한이 부여된 토큰을 자원이 아닌 주체가 갖도록 하여 접근 제어 처리 시간을 감소할 수 있다.

본 논문에서 제안한 capability 토큰에 대한 무결성은 시스템 내부의 인가 스토어를 통하여 검증하였다. 향후 인가된 토큰에 대해 신뢰 할 수 있는 제3의 인증기관(trusted third party)에 대한 문제가 개선되어야할 것으로 보인다. 사물인터넷 환경에서는 정적인 위치에서 사용되는 것이 아니라 동적인 환경이기 때문에 토큰 전달 및 검증 시간의 최소화되어야 한다. 따라서 토큰 경량화를 통해 이와 같은 문제점 개선 방안에 대한 추가연구가 필요하다.

또한, 현재는 구현한 테스트베드를 통하여 사물인터넷 환경에서 다수의 사용자가 capability 토큰을 사용하여 접근제어를 수행하도록 하였으나, 추후 연구를 통하여 웹을 통한 사용자뿐만 아니라 다양한 스마트 기기 및 센서를 추가하여 다양한 장치 및 사물에 대한 Capability 토큰 기반 접근제어를 구현할 것이다.

References

[1] Rolf H. Weber,"Internet of Things - New security and privacy challenges," Computer Law & Security Review, Volume 26, Issue 1, pp. 23-30, Jan. 2010.

[2] S. Gusmeroli, S. Piccione, and D. Rotondi, "IoT access control issues: a capability based approach," IMIS-2012, pp.787 - 792, July 2012.

[3] Gi-hyen Kim, Access Control Technology Overview, Korea Information Security

Agency, June 2011.
[4] L.J.Janczewski and A.M.Colarik, Cyber Warfare and Cyber Terrorism, IGI Global, Hershey,PA, pp.318-326, Sept. 2008.
[5] Pierangela Samarati, Sabrina de Capitani di Vimercati, "Access Control: Policies, Models, and Mechanisms, Foundations of Security Analysis and Design," FOSAD 2000, LNCS 2171, pp.137-196, 2001.
[6] C.P.Pfleeger, Security in Computing, Prentice-Hall, Inc., New Jersey, 1997.
[7] Ministry of Public Administration and Security Department of personal information protection, A homepage privacy exposure protect guide-line, July 2012.
[8] Sergio Gusmeroli, Salvatore Piccione, and Domenico Rotondi, "A capability-based security approach to manage access control in the Internet of Things," Mathematical and Computer Modelling 58, pp.1189-1205, Sept. 2013.
[9] M. Miller, Ka-Ping Yee, and J. Shapiro, "Capability Myths Demolished," Tech.Report SRL 2003-02, Johns Hopkins University, 2003.
[10] L. Fang, D. Gannon, and F. Siebenlist, "XPOLA－an extensible capability based authorization infrastructure for grids," 4th Annual PKI R&D Workshop, pp. 30-40, Apr. 2005.

〈저자소개〉

이 범 기(Bum-Ki Lee) 학생회원
2013년 8월: 국립목포대학교 정보보호학과 학사
2015년 2월: 국립목포대학교 정보보호기술학협동과정 석사
관심분야: NFC 보안, 빅 데이터 보안, 클라우드 컴퓨팅 보안

김 미 선(Mi-Sun Kim) 정회원
1996년 2월: 국립목포대학교 컴퓨터공학과 학사
2000년 2월: 국립목포대학교 컴퓨터공학과 석사
2007년 2월: 국립목포대학교 컴퓨터공학과 박사
2012년 12월~현재: 국립목포대학교 정보보호학과 초빙교수
관심분야: 정보보호, 프로그래밍 언어, 컴퓨터 네트워크, 모바일 시스템 보안

서 재 현(Jae-Hyun Seo) 종신회원
1985년 9월: 전남대학교 계산통계학과 학사
1988년 2월: 중앙대학교 전자계산학과 석사
1996년 8월: 전남대학교 전산통계학과 박사
1996년 9월~현재: 국립목포대학교 정보보호학과 교수
관심분야: 정보보호, 시스템 및 네트워크보안, 컴퓨터 네트워크, 모바일 네트워크 보안