IoF-Cloud 기반 분산된 IoT 장비들을 위한 Download Over-the-Air 기능의 개념 설계

(Concept Design of Download Over-the-Air functions for IoF-Cloud based distributed IoT devices)

차병래*,**, 최명수**, 박선*, 김형균****, 김용일***, 김종원*

(ByungRae Cha, MyeongSoo Choi, Sun Park, HyeongGyun Kim, YongIl Kim, JongWon Kim)

요 약

앞으로 20여 년 동안은 인터넷과 스마트폰에서 사물인터넷으로 대이동이 시작될 것이다. 사물인터넷의 핵심은 인간을 둘러싼 사물들이 서로 연결되면서 인간에게 새로운 편익 혹은 가치를 부여하는 것이다. 미래의 산업 환경은 제품 생산 과정에 있는 모든 기계와 기계, 공장과 공장을 포함한 모든 공정이 긴밀하게 연결되며, 가상 세계와 실세계의 결합인, 전체 제품과 생산주기의 디지털화를 통해 마침내 디지털 공장은 현실로 구현될 것이다. 제안한 IoT 또는 IIoT 기반의 다운로드 OTA (Over-the-Air)는 네트워크를 통한 임의의 형식과 크기의 미디어 객체(Media Object)를 다운로딩하기 위한 유연성 있는 메커니즘을 제공한다. 더불어 제안한 DLOTA는 경량 암호화, OTP, 그리고 CapBAC 기술에 의한 일부분의 보안 기능을 제공한다.

■ 중심어 : 사물 인터넷의 디바이스, IoF-Cloud, OTA(Over-the-Air), 능력기반 접근제어

Abstract

Over the next 20 years it will begin the exodus from the Internet and smart phones to the Internet of Things. The heart of IoT gives new utility and value with connectivity among things around people to the human. In future, Industrial environment will be intimately connect all among machines and machines or factories and factories in all processing, and by digitizing of all goods and production life-cycle, which is a combination of virtual world and real world, the digital factory will become reality eventually. The proposed IoT or IIoT based Download OTA (Over-the-Air) provides a flexible mechanism for downloading Media objects of any type and size from a network. Moreover, proposed IoT based DLOTA provides a part of security by lightweight encryption, OTP, and CapBAC technique.

■ keywords : IoT Device, IoF-Cloud, OTA(Over-the-Air), CapBAC(Capability-based Access Control)

Ⅰ. 서 론

데스크탑이 인터넷에 연결된 이후, 스마트폰에 인터넷이 연결되기까지 대략 20여년이 소요됐다. 최근 몇 년 전부터 알게 모르게 시작된 사물인터넷의 시대는 어느새 우리 곁에 바싹 다가와 있으며, 앞으로 20여 년 동안은 인터넷과 스마트폰에서 사물인터넷으로 대이동이 시작될 것이다. 사물인터넷의 핵심은 인간을 둘러싼 사물들이 서로 연결되면서 인간에게 새로운 편익 혹은 가치를 부여하는 것이다. 최근 구글 이외에도 삼성, 인텔 등 쟁쟁한 기업들이 잇따라 사물인터넷 시장에 뛰어드는 이유는 아마도 성장세가 둔화되고 있는 스마트폰 시장에서 그 답을 찾을 수 있다. 스마트폰이 인간을 중심으로 하여 언제 어디서든 연결될 수 있는 상태를 만들어 주었다면, 사물인터넷은 인간 주변의 모든 사물을 연결하고 인간과 상호 소통할 수 있도록 만드는 것이다. 사물인터넷은 성장 정체기에 접어든 스마트폰을 대신하여 차기 스마트폰 시장 및 새로운 블루 오션을 주도할 것으로 예상된다[1].

또한, 미래의 산업 환경은 인더스트리 4.0을 중심으로 다양한 변화가 주도될 것이며, 인더스트리 4.0으로의 도약을 통해 제품 생산 과정에 있는 모든 기계와 기계, 공장과 공장을 포함한 모든 공정을 긴밀하게 연결될 것이다. 인더스트리 4.0의 핵심 개념은 디지털화이며, 시장 동향은 모듈화, 생산 단계에서의 디지털드로잉, 그리고 기계간의 커뮤니케이션을 지향하고 있다. 가

* 정회원, 광주과학기술원 전기전자컴퓨터공학부
** 정회원, 제노테크(주)
*** 정회원, 호남대학교 인터넷소프트웨어학과
**** 정회원, 한국발명진흥회
이 논문은 2016년도 미래창조과학부의 지원을 받아 수행된 시큐리티 스타트업 R&D 지원 사업의 결과물임 (No. B0717-16-0088).

접수일자 : 2016년 11월 12일
수정일자 : 2016년 12월 23일
게재확정일 : 2016년 12월 26일
교신저자 : 김종원, e-mail : jongwon@smartx.kr

상 세계와 실세계의 결합인, 전체 제품과 생산주기의 디지털화를 통해 마침내 디지털 공장은 현실로 구현될 것이다[2].

Industrial IoT(IIoT)는 에코 시스템을 구성하는 요소들의 확대와 다양한 디바이스에 의하여 보안의 중요성이 증대하고 있다. IIoT의 보안은 정보기술과 운영 기술의 융합에 의한 신뢰성을 확보할 수 있으며, 정보 기술의 신뢰성은 Security, Reliability, Privacy, 그리고 Resilience 측면이며, 운영 기술의 신뢰성은 Safety, Resilience, Reliability, 그리고 Security 측면이며, [그림 1]과 같이 나타낼 수 있다[3]. 또한, IIoT는 시스템 뷰, 가능 뷰, 그리고 보안 뷰 측면에서 end-to-end로부터 클라우드까지 OT와 IT 모두가 연결을 확장하여 기능을 제공하게 되며, [그림 2]와 같이 나타낼 수 있다.

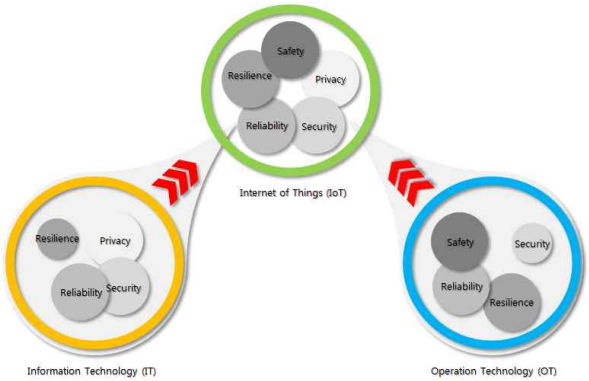

그림 1. IT와 OT 신뢰성의 융합

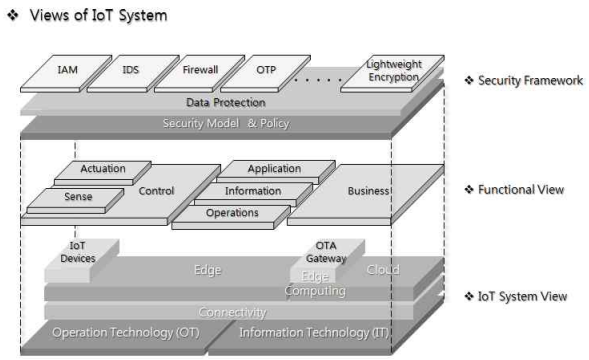

그림 2. IIoT의 시스템/기능/보안 프레임워크의 뷰

본 논문의 관련 연구에서는 OMA의 OTA, SOTA/FOTA, IoT 보안과 CapBAC 접근제어에 관하여 간략하게 기술하며, 3장에서는 IoT 디바이스 기반 IoF-Cloud의 DLOTA 메커니즘을 설계 및 기능들을 제안하고자 하며, 4장에서는 DLOTA 메커니즘과 관련된 보안 문제를 기술한다. 그리고 마지막으로 결론과 향후 연구의 내용을 기술한다.

Ⅱ. 관련 연구

1. OMA의 OTA, 그리고 SOTA/FOTA

OMA (Open Mabile Alliance)의 Download OTA (Over-the-Air)는 네트워크를 통한 임의의 형식과 크기의 미디어 객체(Media Object; MO)를 다운로딩하기위한 유연성 있는 메커니즘을 제공한다[4, 5]. [그림 3]과 [그림 4]는 OMA의 OTA 기능적 아키텍처와 다운로드 절차를 나타낸 것이다.

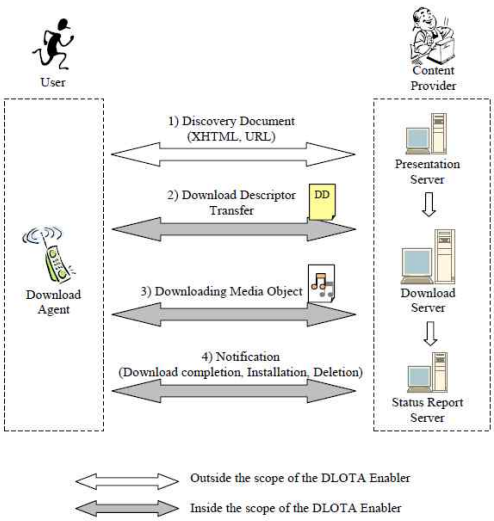

그림 3. OMA의 OTA 기능적 아키텍처

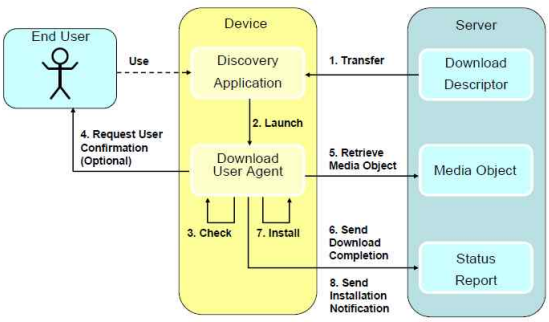

그림 4. OMA의 다운로드 절차

더불어, 최근 OTA 기술은 Connected Care 분야에서 세부화되어 SOTA(Software Over-the-Air)와 FOTA(Firmware Over-the-Air)로 구분되고 있으며, [그림 5]는 모바일과 자동차 분야의 FOTA/SOTA 기술의 비교를 나타내고 있으며[6], 네트워크 장비 분야에서는 Zero-Touch Provisioning 기술[7, 8]이 각광을 받고 있으며, [그림 6]은 시스코의 ZTP의 절차를 나타낸 것이다.

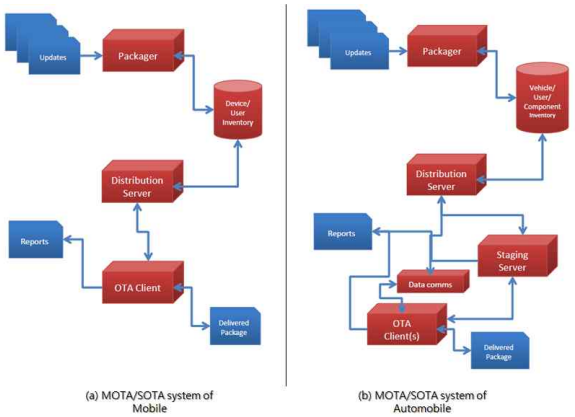

그림 5. 모바일 및 자동차 분야의 OTA 기술 비교 [6]

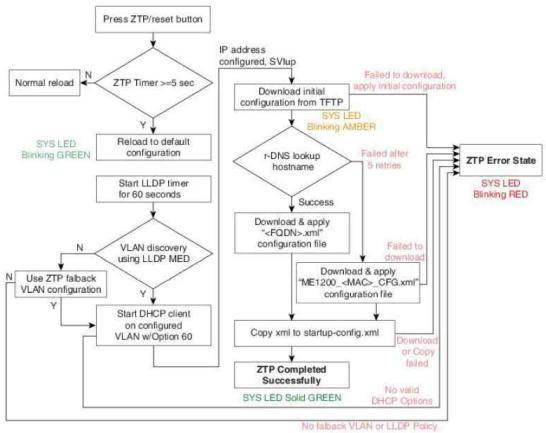

그림 6. 시스코의 ZTP의 Activation 절차 [7]

2. 다양한 IoT 보안과 CapBAC 접근제어

Wind River Systems[9]은 IoT 분야의 모든 가능한 사이버 위협을 효과적으로 경감시키기 위한 완전한 해결책은 없으며, 과거로부터 유효성이 증명된 IT 보안 제어 기술을 통하여 IoT의 고유한 제약 사항과 미래 네트워크에 효과적으로 진화해야 한다고 말하고 있으며, 5 가지 핵심 사항을 다음과 같이 정의하였다. 그리고 [그림 7]은 일반적인 IoT 토폴로지를 나타낸 것이다.

- The evolution of network security
- New threats, constraints, and challenges
- Building security in from the bottom up
  1. Secure booting
  2. Access control
  3. Device authentication
  4. Firewalling and IPS
  5. Updates and patches
- IT starts in the OS
- The end-to-end security solution

Tripwire[10]의 Industrial IoT (IIoT)의 보안을 위한 5 가지 핵심 도전 사항을 다음과 같이 정의 및 나열하였으며, 나열된 핵심 도전들은 IIoT 구현의 어려움에 직면하게 하며, 담당 부서 및 기업, 그리고 제조업체들 모두가 향후 기술의 새로운 동향을 탐색해야함을 의미한다.

- Key Challenge #1: Setting on device capabilities
- Key Challenge #2: Supply chain concerns
- Key Challenge #3: Security
- Key Challenge #4: Bridging the gaps that divide us
- Key Challenge #5: Safety

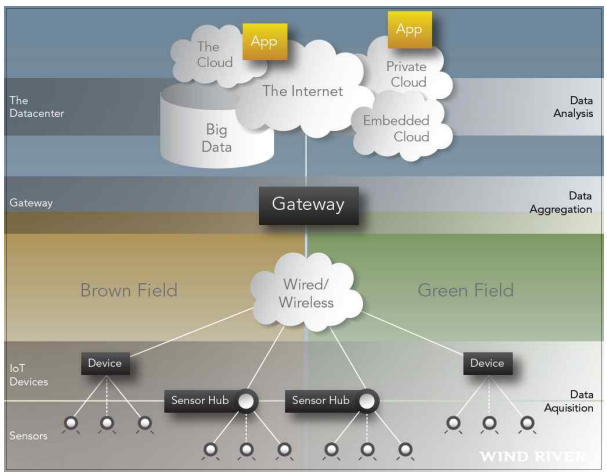

그림 7. 일반적인 IoT 토폴로지 [10]

컴퓨팅 자원에 대한 접근 제어는 보안의 핵심 주체중의 하나이며, 접근 통제는 어떤 사용자들이 어떤 파일 또는 서비스에 접근할 수 있는 가를 통제하는 것 이상의 것들을 다룬다. 객체에서 주체로의 정보 전송을 접근이라고 하며, 주체는 접근을 통하여 수동적 실재, 즉 객체에 대한 정보 또는 객체로부터 데이터를 요구하는 능동적 실재이다. 접근 제어는 객체의 기밀성(confidentiality), 무결성(integrity), 그리고 가용성(availability)을 보호하는 데 필요하다.[11] 접근제어 기법으로는 접근제어 리스트(Access Control List, ACL), 역할기반 접근제어(Role BasedAccess Control, RBAC), 속성기반 접근제어(Attribute Based Access Control, ABAC) 등이 있으며, 최근에는 IoT를 위한 접근제어 기법으로 Capability 기반 접근 제어, 위치-시간기반 접근제어에 대한 연구가 진행되고 있다.

RBAC(role based access control)과 ABAC (attribute based access control)와 같은 권한 부여 프레임워크는 많은

상호운용성을 지원이 필요한 분산 환경에서는 서비스의 확장성, 관리성, 그리고 효율성 등을 효과적으로 지원하지 못한다[12]. [그림 8]은 ACL(access control list)과 CapBAC (capability based access control)과의 개념을 비교하여 위하여 다이어그램으로 나타낸 것이며, CapBAC는 ACL과는 반대로 객체 측면에서 수동적인 접근 제어를 갖는 것이 아니라, 주체 측면에서 능동적인 접근 제어를 갖는 개념이다.

Capability 기반 접근제어는 주체가 권한 리스트를 가지고 있으며, 자신이 갖고 있는 Capability를 서비스 제공자에게 제시하면, 서비스 제공자는 Capability를 확인하여 인가한다. 반복적인 서비스 요청이 발생할 경우 ACL 기반 시스템에서는 반복적으로 인증 프로세스가 진행되지만, Capability 기반 접근제어는 기발행된 Capability 를 통해 반복 작업을 최소화 할 수 있으며, 따라서 ACL 기반 시스템에 비해 워크플로우가 가볍다고 할 수 있다[12, 13].

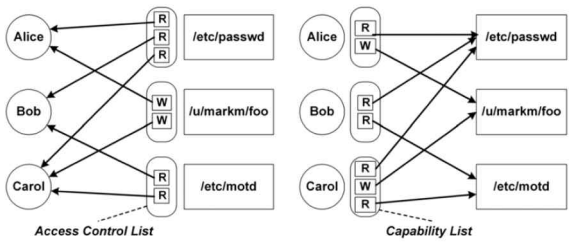

그림 8. ACL과 CapBAC의 개념 비교

III. IoT 기반 DLOTA 메커니즘 설계

IoT(Internet of Things)는 물건을 서로 연결하면 새로운 가치를 창출한다는 개념이며, Things의 사전적 의미로는 물질적인 물건만이 아니라 무형의 일까지 포함하는 단어이다[14]. 유형의 물건에서 무형인 서비스까지 포함하는 사물로 변화 및 진화를 거듭하고 있는 상황이다. 이러한 개념을 농가로 확대하기 위하여 IoF(Internet of Farming)-Cloud를 정의하고자 한다.

IoF-Cloud의 개념 및 정의는 스마트 팜을 365일 안정적으로 운영할 수 있도록 클라우드 기반으로 스마트 팜을 확장하는 것으로 정의할 수 있으며, 즉 1세대 스마트 팜에 지능제어 알고리즘 기반 복합환경 제어, 클라우드 기반 빅데이터 분석/영농의사결정, 복합 에너지 관리, 지능형 자동화 농기계 기반 스마트 농작업을 지원할 수 있도록 구성하는 것으로 정의한다. 그리고, [그림 9]는 애로사항 “네트워크/전원 단절, 시스템 오류 등의 상황에서 시스템의 365일 안정적 운용”과 애로사항 “측정된 데이터의 활용이 미흡”을 해결하기 위한 IoF-Cloud의 개념도와 실환경 테스트베드를 나타낸 것이다[15].

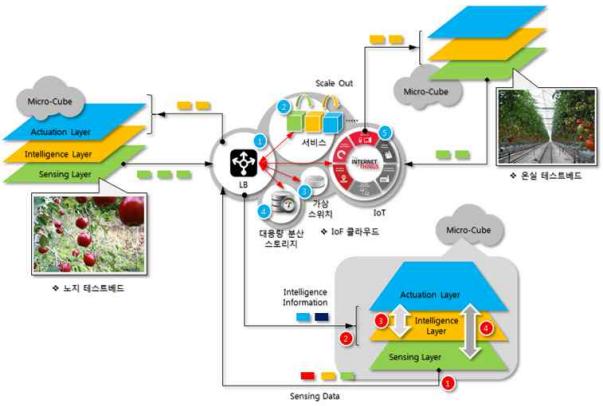

그림 9. IoF-Cloud의 개념도와 실환경 테스트베드

스마트 팜의 실사용자들의 애로사항들을 해결하기 위하여 [그림 9]의 IoF 클라우드는 퍼블릭 클라우드 기반의 영농 빅데이터를 수집 및 저장, 그리고 분석을 통한 영농 의사 결정을 지원하게 된다. IoF-Cloud의 안정성 향상을 위하여 로컬 영역에 MicroCube에 의한 시스템의 이중화 구축 및 서비스를 지원하게 된다. IoF 클라우드는 LB, 서비스, 가상 스위치, 대용량 분산 스토리지, 그리고 IoT 모듈들로 구성될 것이며, MicroCube들에서 전송된 데이터와 서비스로 제공되는 분석 결과를 MicroCube의 Intelligence Layer에 전송 등의 다양한 서비스들을 중앙에서 관리 및 감독하게 된다. 또한, MicroCube는 3계층으로 Sensing Layer, Intelligence Layer, 그리고 Actuation Layer로 구성된다. MicroCube는 퍼블릭 클라우드에 구축된 IoF 클라우드의 Edge Computing 역할을 수행함과 동시에 센싱된 데이터의 전송 및 백업 기능을 지원하게 된다. Micro-Cube의 Sensing Layer는 노지 또는 온실의 테스트베드에서 작물의 생육 환경 정보를 센싱하여 MicroCube의 데이터 저장소와 IoF 클라우드로 데이터를 전송하게 되며, [그림 9]의 ❶과 같이 나타낸다. MicroCube의 Intelligence Layer는 [그림 9]의 ❷와 같이 IoF 클라우드로 부터 전송된 데이터를 수신하며, 전송된 명령을 해석하여 [그림 9]의 ❸과 같이 MicroCube의 Actuation Layer에 명령을 하달하게 된다. 마지막으로 MicroCube의 Actuation Layer는 Intelligence Layer에서 하달된 명령에 따라 다양한 액션을 수행하게 되며, [그림 9]의 ❹와 같이 나타낸다. 이러한 절차의 결과는 다시 [그림 9]의 ❶과 같이 반복되며 다양한 액션들이 순환적으로 운용하게 될 것이다.

1. IoT 디바이스 기반 DLOTA

본 논문에서 제안하는 IoT 또는 IIoT의 DLOTA는 [그림 10]에 나타낸 것과 같이 개발자들이 모든 종류의 IoT 디바이스들을 하나씩 하나씩 설치 및 업그레이드 하는 복잡한 문제들을 간편

하게 해결할 수 있으며, 다양한 영역에 적용이 가능하게 된다. [그림 11]은 제안하는 IoT 디바이스 기반 IoF-Cloud의 DLOTA 메커니즘의 환경 다이어그램을 나타내며, [표 1]은 DLOTA 메커니즘의 전달 방법을 분류 및 Use Case들을 정의하였다.

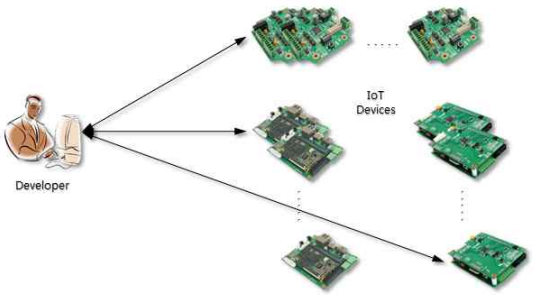

그림 10. 일반적인 IoT 디바이스의
SW 유지/보수 개념

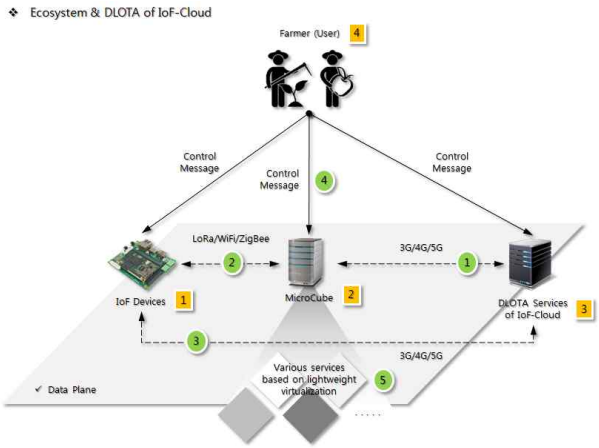

그림 11. IoT 디바이스 기반 IoF-Cloud의 DLOTA
메커니즘의 환경 다이어그램

표 1. IoT 디바이스 기반 DLOTA 메커니즘의 전달 방법

		MO delivery method	
		Pull	Push
DD delivery method	Pull	Pull-Pull (Usecase 1)	Pull-Push (Usecase 2)
	Push	Push-Pull (Usecase 3)	Push-Push (Usecase 4)

Usecase 1: pull-pull scenario는 기본적인 DLOTA 절차가 해당되며, Usecase 2: pull-push scenario는 Broadcast 프로토콜의 경우이며, 그리고 Usecase 4: push-push scenario는 SIAD를 의미한다.

2. IoT 기반 DLOTA 메커니즘의 기능적 아키텍처와 DD/MO

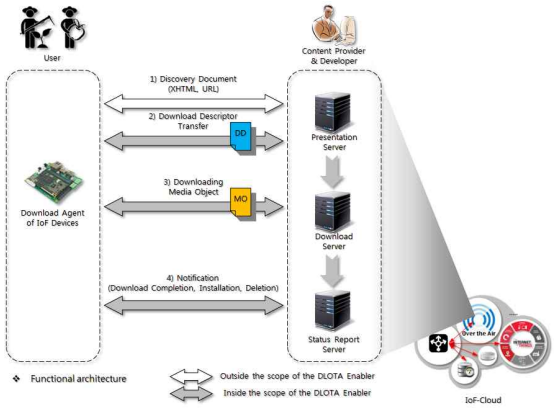

그림 12 . IoT 디바이스 기반 DLOTA 메커니즘의
기능적 아키텍쳐

IoT 기반의 DLOTA 메커니즘은 OMA의 DLOTA와 유사하게 Download Agent와 Server 간의 DD (Download Description)와 MO (Media Object) 정보에 의해서 동작하게 되며, DD는 MO를 기술하기 위한 요소들로 구성되었으며, DD의 내용을 참조하여 MO를 다운로드 및 설치하게 되며, [그림 12]는 IoT 기반 DLOTA 메커니즘의 기능적 아키텍처를 나타낸다. Content Provide & Developer는 클라우드 자원(IoF-Cloud)을 서버 가상화 기술에 의하여 가상 머신 위에 Present Server, Download Server, 그리고 Status Report Server 등으로 구성하게 된다. 특히, Ubuntu Linux의 경우에는 모든 타입의 컨테이너들(process container Docker, machine container LXD, 그리고 application container Snapd)의 지원이 가능하다. 사용자는 Present Server로부터 DD를 전송받으며, Download Server로부터는 MO를 전송 받으며, 다운로드 및 설치가 완료되면, Status Report Server로 상태 정보를 전송하게 된다.

IoT를 지원하기 위한 클라우드 기반 DLOTA 서비스 설계의 다이어그램을 [그림 13]과 같이 표현한다. IoT 디바이스와 IoT Gateway(MicroCube)에 존재하는 Discovery Application은 클라우드의 가상머신인 Present Server로부터 DD를 다운받고 내용을 분석하게 되며([그림 13]의 ① 참조), DD에는 CapBAC 접근제어 정보를 포함하게 된다.

IoT Gateway는 DD 정보에 의해서 IoT 디바이스들을 위한 MO 정보를 획득함과 동시에 Download User Agent 모듈에 설치하게 된다([그림 13]의 ② 참조). Download User Agent는 MO의 정보와 CapBAC 접근 제어에 의한 점검을 하게 되며([그림 13]의 ③ 참조), 특이 사항이 존재하지 않는 정상 상황과 CapBAC 접근제어에 문제가 발생하지 않는다면 클라우드 기반

의 OTA 서비스로부터 해당하는 MO를 검색 및 다운로드하게 된다([그림 13]의 4 참조). 최종 사용자의 승인([그림 13]의 5 참조)에 의하여 다운로드의 권한을 갖게 되며, 다운로드된 MO는 IoT Gateway의 스토리지에 저장([그림 13]의 6 참조)하게 된다. IoT Gateway는 IoT 디바이스들을 위한 스토리지 캐쉬 기능을 수행함과 동시에 3G/4G/5G의 글로벌 트래픽을 LPWAN(Low power wide area network) 등의 로컬 트래픽으로 전환하는 역할을 수행하게 된다.

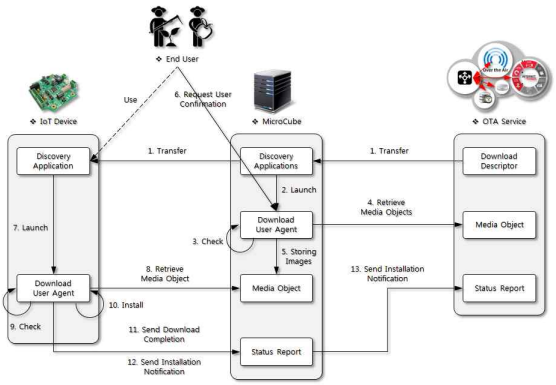

그림 13. 클라우드 기반의 DLOTA 서비스 설계

최종 사용자인 말단의 IoT 디바이스는 자체의 타입에 해당하는 MO 정보를 획득함과 동시에 Download User Agent 모듈에 설치하게 된다([그림 13]의 7 참조). Download User Agent는 IoT Gateway로부터 MO의 검색 및 다운로드를 진행하게 된다([그림 13]의 8 참조). Download User Agent는 MO의 정보와 CapBAC 접근 제어에 의한 점검 및 설치를 진행하게 된다([그림 13]의 9와 10 참조). IoT 디바이스에 MO의 설치가 완료되면 IoT Gateway에 다운로드 및 설치 완료의 상태 정보를 전송하게 된다([그림 13]의 11과 12 참조). 이러한 정보에 의해서 IoT 디바이스의 상태 및 이력 정보를 관리하게 된다. 또한 IoT Gateway는 클라우드 기반의 OTA 서비스의 Status Report Server로 요약된 상태 정보를 전송하게 된다([그림 13]의 13 참조).

3. 기본 DLOTA

기본적인 DLOTA의 절차는 [그림 14]와 같이 나타낼 수 있으며, Discovery Application에 의해서 DD와 MO 컨텐츠를 발견하게 된다([그림 14]의 ① 참조). 이때, IoT 디바이스는 IoT Gateway로부터 DD를 요청하게 되며([그림 14]의 ② 참조), IoT Gateway는 요청된 DD를 전송하게 된다([그림 14]의 ③ 참조). IoT 디바이스는 IoT Gateway로부터 전송된 DD에 의해서 다운로드할 MO의 정보와 IoT 디바이스의 Capability를 점검하게 되며, 더불어 CapBAC 접근제어가 진행하게 된다([그림 14]의 ④ 참조). IoT 디바이스는 사용자에게 승인을 요청하며, 사용자의 승인에 의하여 MO의 다운로드를 진행할 권한을 갖게 된다([그림 14]의 ⑤와 ⑥ 참조). IoT 디바이스는 IoT Gateway에게 MO를 요청하면 IoT Gateway는 요청한 MO를 전송하게 된다([그림 14]의 ⑦와 ⑧ 참조). MO의 전송이 완료되면 IoT 디바이스는 MO를 이용하여 펌웨어 및 소프트웨어를 설치([그림 14]의 ⑨ 참조)를 하게 되며, 설치가 완료된 시점에 IoT Gateway에게 설치 완료된 상태 정보를 전송하게 된다([그림 14]의 ⑩ 참조).

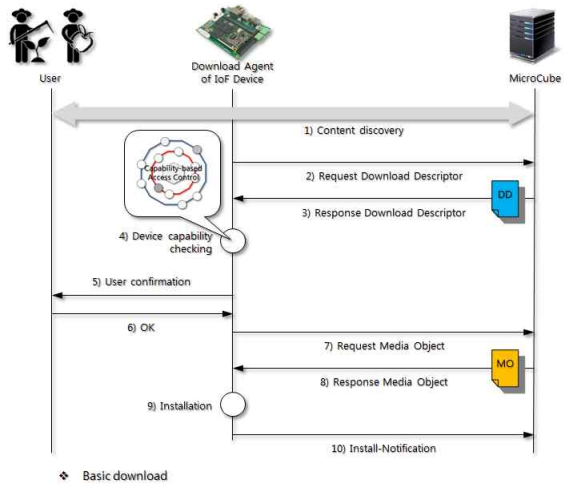

그림 14. 기본적인 DLOTA 절차

4. Media Object의 업데이트와 삭제

기존 펌웨어 또는 소프트웨어를 최신 버전으로 업데이트를 위한 절차와 제거 절차를 [그림 15]와 [그림 16]에 나타내며, Discovery Application에 의해서 기존 MO의 버전 정보보다 최신 버전의 MO 정보를 얻게 된다.

IoT 디바이스는 IoT Gateway에게 최신 버전의 DD 정보를 요청하게 되며, IoT Gateway는 요청된 DD를 IoT 디바이스에게 전송하게 된다([그림 15]의 ①과 ② 참조). 전송된 DD를 분석하여 펌웨어 또는 소프트웨어의 업데이트 및 Capability를 점검하게 되며([그림 15]의 ③ 참조), 펌웨어 또는 소프트웨어의 업데이트를 버전 정보와 Capability 에 의해서 업데이트 진행이 타당하다면 최종 사용자에게 승인을 요청하며, 승인에 의하여 권한을 갖게 된다([그림 15]의 ④와 ⑤ 참조). IoT 디바이스는 권한 획득하게 됨과 동시에 IoT Gateway에게 새로운 버전의 MO를 요청하게 되며, IoT Gateway는 요청된 MO를 전송하게 된다([그림 15]의 ⑥과 ⑦ 참조). IoT 디바이스는 다운로드된 MO를 이용하여 설치 과정이 진행되며([그림 15]의 ⑧ 참조), 설치 완료시에 IoT Gateway에게 상태 정보를 전송하게 된다([그림 15]의 ⑨ 참

조).

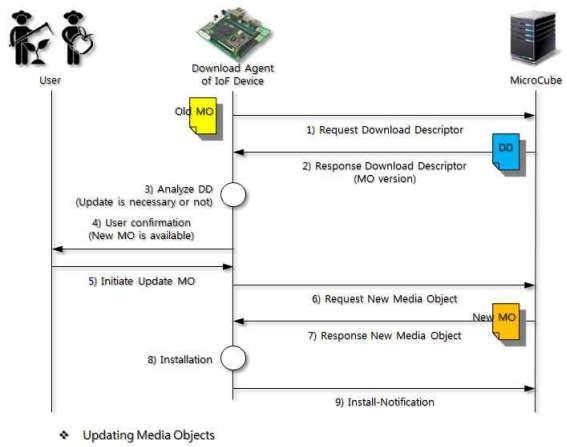

그림 15. Media Object의 업데이트 절차

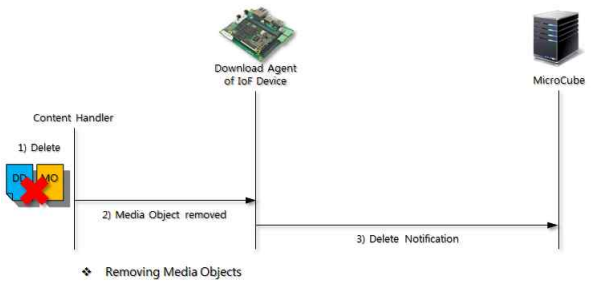

그림 16. Media Object의 제거 절차

[그림 16]은 기존의 MO를 제거하는 과정을 나타내며, IoT 디바이스의 Content Handler는 신규 펌웨어 또는 소프트웨어가 설치됨에 따라 기존의 DD와 MO를 제거([그림 16]의 ①과 ②참조)하게 되며, IoT 디바이스는 IoT Gateway에 기존의 MO가 제거된 상태 정보를 공지하게 된다([그림 16]의 ③ 참조).

5. Broadcast 프로토콜 기반의 DLOTA

Broadcast 프로토콜 기반의 DLOTA는 긴급한 패치가 필요하거나 또는 IoT Gateway에 문제가 발생한 경우하여 제 역할을 수행하지 못하는 상황에 사용되며, OTA 서비스를 지원하는 클라우드에서 모든 연결된 IoT 디바이스에 전송하는 절차를 의미하며, [그림 17]에 나타낸다. 특히 모든 IoT 디바이스가 연결되므로 네트워크 트래픽의 폭주를 예방하기 위하여 다운로드를 위한 후보 시간 리스트를 DD에 삽입하여 전송하게 된다([그림 17]의 ③ 참조). IoT 디바이스는 후보 시간 리스트에서 다운로드 시간 정보를 획득하며, 정해진 시간 대역에 DLOTA 서비스를 지원받게 되며([그림 17]의 ④ 참조), 설치가 완료되면 상태 정보를 전송하게 되며, 전송된 정보에 의해서 MO 설치가 실패하거나 DLOTA 서비스가 지원되지 않은 다른 IoT 디바이스들을 다시 스케쥴링하게 된다.

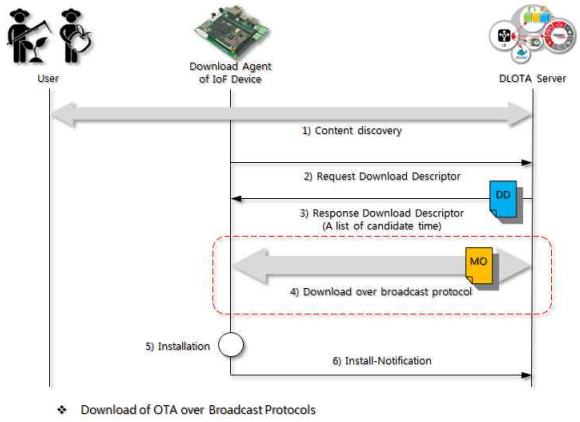

그림 17. Broadcast 프로토콜에 의한 DLOTA 절차

6. 시간 예약 기능의 DLOTA

Broadcast 프로토콜 기반의 DLOTA는 DLOTA 서비스를 지원하는 서버에서 서비스 시간을 일방적으로 계획 및 진행한다면, 반대로, 사용자 측에서 다운로드 시간을 예약하여 다운로드 서비스를 진행하는 절차를 나타내며, [그림 18]과 같이 나타낸다.

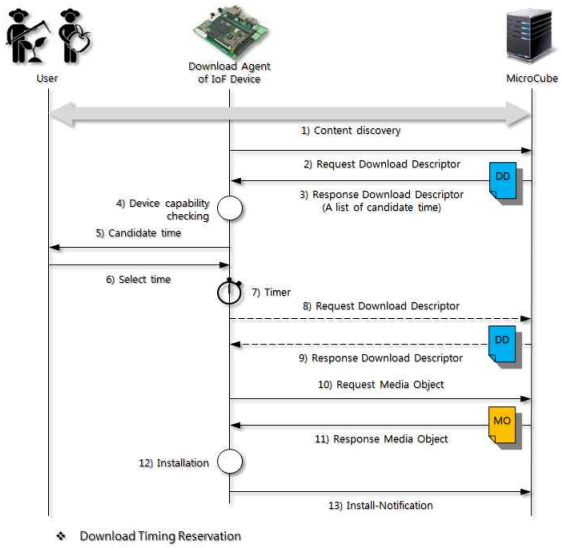

그림 18. 다운로드 시간 예약에 의한 DLOTA 절차

기본적인 DLOTA의 [그림 14]와 절차는 동일하며, 최종 사용자가 DLOTA 서비스 시간을 설정하는 것과 DD의 재전송 절차만 차이가 있다. 최종 사용자가 서비스를 위한 후보 시간 리스트

정보를 전달 받으며, 후보 시간 리스트에서 원하는 시간 대역을 사용자가 설정하게 된다([그림 18]의 ⑤와 ⑥ 참조). 사용자의 시간 설정에 의하여 자동적으로 타이머가 동작하게 되며([그림 18]의 ⑦ 참조), 예약된 시간 대역에서 IoT 디바이스는 IoT Gateway에게 DD의 재전송을 요청하게 된다([그림 18]의 ⑧과 ⑨ 참조). 예악 시간까지의 다운로드할 MO의 변경을 탐지하기 위한 절차이며, 변경되지 않았다면, IoT 디바이스는 MO을 요청하게 되며, IoT Gateway는 요청된 MO를 전송하게 된다. 다운로드한 MO를 이용하여 설치가 진행되고, 설치 완료를 IoT Gateway에게 공지하게 된다.

7. SIAD 기능의 DLOTA

SIAD(Server Initiated Automatic Download)는 DLOTA 메커니즘의 전달 방법의 Usecase 4: Push-Push 시나리오에 해당하는 모델이며, [그림 19]와 같이 나타낸다. 시스템 SW 측면 또는 신규 IoT 디바이스들이 출시하는 경우에 DLOTA 서버에서 일방 또는 강제적으로 다운로드를 실행하게 된다.

클라우드 기반의 DLOTA 서비스는 IoT Gateway와 IoT 디바이스들에게 일방적으로 DD를 전송([그림 19]의 ② 참조)하게 되며, DD의 기술된 정책과 보안에 의해서 MO를 다운로드 및 설치하게 된다([그림 19]의 ⑤와 ⑥ 참조). 설치 완료시에 클라우드 기반의 DLOTA 서비스의 Status Report Server에 상태 정보를 전송하게 된다.

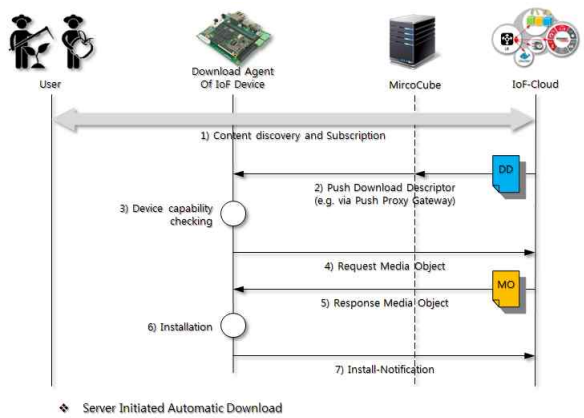

그림 19. SIAD 절차

Ⅳ. 결 론

사물인터넷의 핵심은 인간을 둘러싼 사물들이 서로 연결되면서 인간에게 새로운 편익 혹은 가치를 부여하는 것이다. 미래의 산업 환경은 제품 생산 과정에 있는 모든 기계와 기계, 공장과 공장을 포함한 모든 공정이 긴밀하게 연결되며, 가상 세계와 실세계의 결합인, 전체 제품과 생산주기의 디지털화를 통해 마침내 디지털 공장은 현실로 구현될 것이다.

제안한 IoT 또는 IIoT 기반의 Download OTA (Over-the-Air)는 네트워크를 통한 임의의 형식과 크기의 미디어 객체(Media Object)를 다운로딩하기위한 유연성 있는 메커니즘을 제공한다. 본 논문에서는 OMA의 OTA를 기반으로 IoT를 위한 DLOTA를 설계 및 구성하였으며, 기본 DLOTA와 Media Object의 업데이트와 삭제 절차, Broadcast 프로토콜 기반의 DLOTA, 시간 예약 가능한 DLOTA 절차 그리고 SIAD 기능의 절차를 세부적으로 설계 및 기술하였다. 더불어 제안한 DLOTA는 경량 암호화, OTP, 그리고 CapBAC 기술에 의한 일부분의 보안 기능을 제공한다.

References

[1] 커넥팅랩, “클라우드와 빅데이터를 뛰어넘는 거대한 연결,” 미래의 창, 2014.
[2] 한석희, 조형식, 홍대순, “미래를 결정지을 제4차 산업 혁명 - 인더스트리 4.0,” 페이퍼로드, 2015.
[3] Industrial Intermnet Consortium, "Industrial Internet of Things Volume G4: Security Framework," 2016.
[4] OMA, "Download Over the Air Specification," Sep., 2006.
[5] OMA, "Download Over the Air Architecture," Aug, 2006.
[6] Roger Hampel, "Keeping the connected car current with SOTA/FOTA," Automative Linux Summit, Sep. 19, 2012. [Internet] https://events.linuxfoundation.org/images/stories/pdf/als2012_hampel.pdf
[7] Cisco Zero Touch Provisioning, [Internet] http://www.cisco.com/c/en/us/td/docs/switches/metro/me1200/controller/guide/b_nid_controller_book/b_nid_controller_book_chapter_011.pdf
[8] Arista Zero Touch Provisioning, [Internet] https://www.arista.com/assets/data/pdf/TechBulletins/Tech_bulletin_ZTP.pdf
[9] Wind River Systems, "White Paper: Security in the Internet of Things - Lessons from the Past for the Connected Future," 2015.
[10] Tripwire, "5 Key Challenges for the Industrial Internet of Things (IIoT)," [Internet] https://www.tripwire.com/state-of-security/featured/5-key-challenges-for-the-industrial-internet-of-things-iiot/
[11] Ed Tittel, "CISSP: Certied Information Systems Security Professional Study Guide," 정보문화사, 2004.

[12] Sergio Gusmeroli, Salvatore Piccione, Domenico Rotondi, “A capability-based security approach to manage access control in the Internet of Things,” Mathematical and Computer Modeling 58, Elsevier, 2013, p.1189-1205.
[13] Jose L. Hernandez-Ramos, Antonio J. Jara, Leandro Marin, and Antonio F. Skarmeta, "Distributed Capability-based Access Control for the Internet of Things," Journal of Internet Services and Information Security (JISIS), Vol. 3, no3/4, 2013, p.1-16.
[14] 오가사하라 오사무, “메이커스 진화론,“ 더숲, 2016년 5월.
[15] 차병래, 최명수, 김봉국, 전오성, 한태호, 김종원, 박 선, “차세대 IoF-Cloud 기반 스마트 온실 및 서비스 연구,” 한국스마트미디어학회 스마트미디어저널, Vol.5, No.3, p.17-24.

저 자 소 개

차병래

2004년 목포대학교 대학원 컴퓨터공학과 졸업(공학박사)
2005년 호남대학교 컴퓨터공학과 전임강사
2009년 ~ 현재 광주과학기술원 전기전자컴퓨터공학부 연구조교수
2012년 ~ 현재 제노테크(주) 대표
<주관심분야 : 정보보안, IDS, Neural Network, Cloud Computing, VoIP, NFC 등>

최명수

2009년 목포대학교 전자공학과 공학박사
2009년 목포대학교 해양텔레매틱스기술개발센터 박사후연구원
2010년 목포대학교 정보산업연구소 연구전임교수
2015년 ~ 현재 제노테크(주) 기업부설연구소 연구소장
<주관심분야 : IoT, Neural Network, Cloud Computing, VoIP, NFC>

박 선

2007년 인하대학교 컴퓨터정보공학과 공학박사
2008년 호남대학교 컴퓨터공학과 전임강사
2010년 전북대학교 인력양성사업단 박사후 과정
2010년 목포대학교 정보산업연구소 연구전임교수
2013년 ~현재 광주과학기술원 전기전자컴퓨터공학부 연구조교수
<주관심분야 : 정보검색, 데이터마이닝, 해양IT정보융합, 클라우드 컴퓨팅, IoT, 스토리지 시스템>

김형균

2004년 조선대학교 대학원 컴퓨터공학과 졸업(공학박사)
2005년 동강대학교 컴퓨터정보과 초빙전임강사
2016년 ~ 현재 한국발명진흥회 전문위원
<주관심분야 : IoT, Neural Network, Cloud Computing, >

김용일

1984년 3월 : 전남대학교 계산통계학과 (이학사)
1986년 2월 : 한국과학기술원 전산학과 (공학석사)
1986년 3월~1994년 2월 : 한국원자력연구소 선임연구원
1994년 3월~2000년 2월 : 초당대학교 컴퓨터학과 조교수
2002년 3월~현재 : 호남대학교 인터넷콘텐츠학과 부교수
<주관심분야 : 빅데이터, 지능형 정보검색, 지능형 에전트>

김종원

1997년 University of Southern California 연구 조교수
1999년 Technology Consultant for VProtect Systems Inc.
2000년 Technology Consultant for Southern California Division of InterVideo Inc.
2001년 광주과학기술원 전기전자컴퓨터공학부 교수
2008년 ~ 현재 광주과학기술원 전기전자컴퓨터공학부 교수
<주관심분야 : Future Internet, SDN & NFV, SDI>