Chapter 02

사물인터넷 부트스트래핑 보안 기술

김영호_한국전자통신연구원 책임연구원
이윤경_한국전자통신연구원 책임연구원
김정녀_한국전자통신연구원 책임연구원

I. 서론

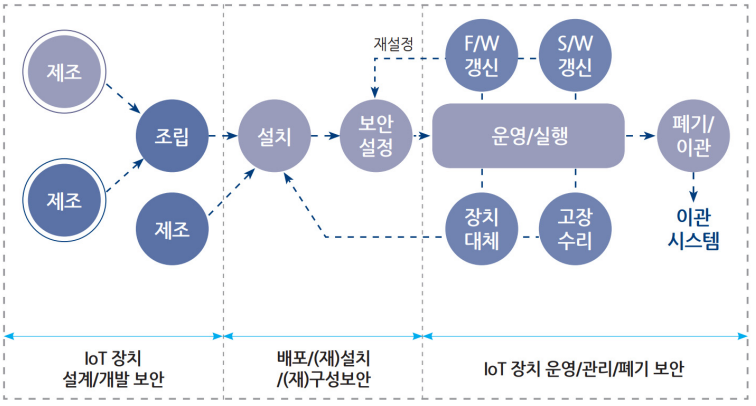
부트스트랩(bootstrap)의 사전적 의미는 신발(boot) 뒤에 달린 손잡이(strap), 또는 스스로 전개하기(development)이다. 따라서 부트스트래핑은 시스템이 정상 동작하기까지 스스로 전개하는 '도움닫기' 과정으로 이해될 수 있다. 컴퓨터 전원이 켜지는 순간 바로 진행되는 부트스트래핑 과정이 대개 보이지도 않고 빠르게 진행되기 때문에 큰 의미 없이 여겨진다.

사물인터넷은 데이터 중심의 정보 기술(Information Technology: IT)과 실세계의 정보 수집과 기기 제어를 담당하던 운영 기술(Operational Technology: OT)을 결합하여 사이버 세계와 물리적인 세계를 연결하는 새로운 서비스를 창출한다[1]. 따라서 사물인터넷 환경에서 부트스트래핑은 이미 잘 정립된 IT 인프라의 신뢰(Root of Trust)로부터 출발하여 OT 인프라로 신뢰 사슬(Chain of Trust)을 확장해 나가는 보안의 도움닫기 과정으로 여겨진다. 최근 5G부터 LPWAN(low-power wide-area network) 서비스에 이르기까지 기기 및 통신방식의 다양성(diversity)과 이질성(heterogeneity)으로 대표되는 사물인터넷 환경에서 기기 스스로 주변의 검증된 네트워크에 접속하고 서비스를 안전하게 준비하는 부트스트래핑의 역할과 범위는 점차 확대되는 추세이다. 본 고에서는 부트스트래핑 기술의 역할 및 범위를 보안 관점에서 살펴보고자 한다. 특히, 사물인터넷 환경에서 추가적으로 요구하는 특성과 대표적인 기술 사례를 통해 발전 방향을 살펴보고자 한다.

* 본 내용은 김영호 책임연구원(☎ 042-860-4891, wtowto@etri.re.kr)에게 문의하시기 바랍니다.
** 본 내용은 필자의 주관적인 의견이며 IITP의 공식적인 입장이 아님을 밝힙니다.
***본 과제는 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. 2018-0-00231, (IoT 2세부) IoT 인프라 공격 확산 방어를 위한 상황 적응형 보안 자율제어 기술개발).

II. 부트스트래핑 기술 소개

부트스트래핑의 기술적인 정의는 “어떤 장치가 실제로 운용 가능한 상태에 도달하기 이전에 취해지는 모든 조치”를 의미한다[2]. 이런 해석을 OT 인프라를 포함하는 사물인터넷으로 확장하면 부트스트래핑 동작은 [그림 1]의 “배포/(재)설치/(재)구성보안” 단계에서 발생한다.

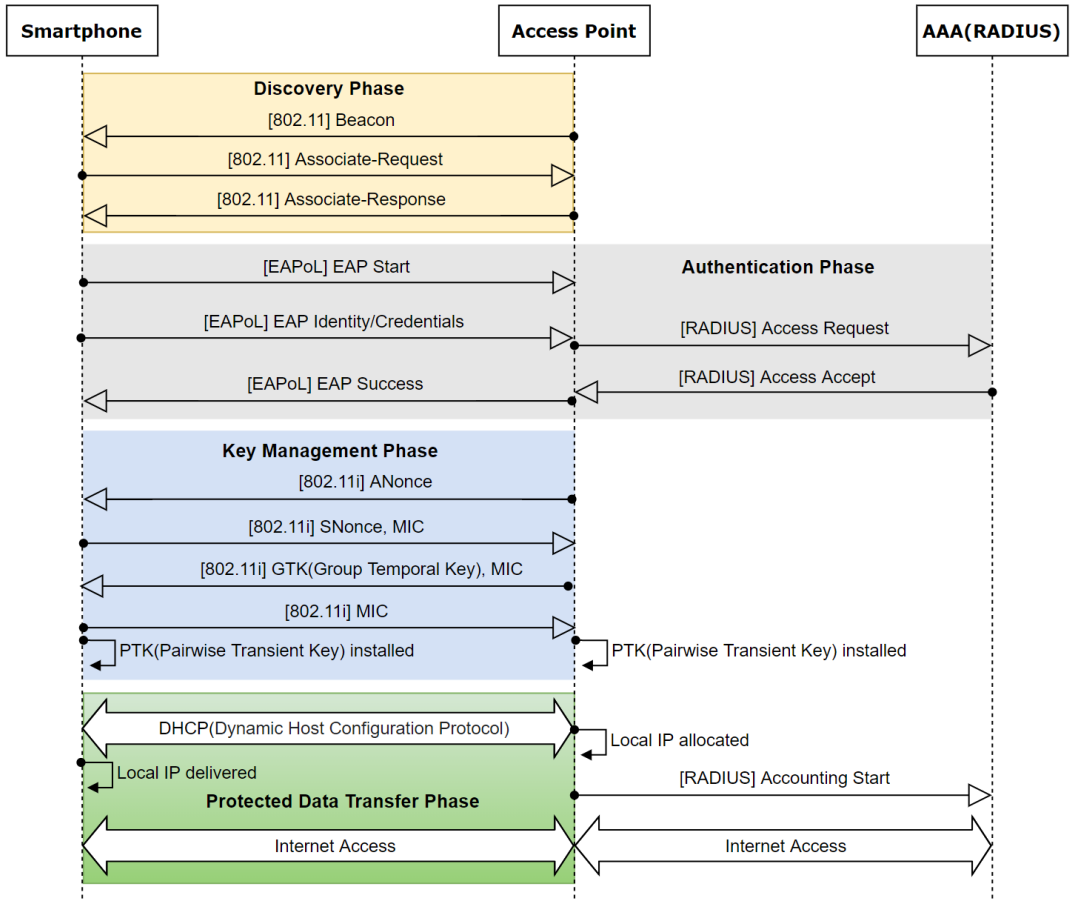


〈자료〉 “IoT 공통 보안 원칙 v1.0”, 사물인터넷 보안 얼라이언스, 2016. 6, p.6.

[그림 1] 사물인터넷 장치의 전주기 단계별 보안 고려사항

따라서 기기가 적대적인 OT 인프라 환경에 노출되더라도 외부와의 신뢰 관계를 확보하기 위한 첫걸음으로 부트스트래핑 기술이 사용될 수 있다. 동일한 의미로 초기화(initialization), 프로비저닝(provisioning), 온보딩(onboarding), 커미셔닝(commissioning) 등 다양한 용어를 사용하고 있다. 우리에게 좀 더 익숙한 IT 인프라의 예를 들면, 스마트폰 사용자가 인터넷을 사용하기 위해서 무선 라우터인 AP(Access Point)를 선택하고 아이디와 비밀번호를 입력하는 등의 네트워크 설정 행위도 안전하게 인터넷 서비스에 연결하는데 필요한 준비과정이기 때문에 넓은 의미에서 부트스트래핑으로 간주할 수 있다.

비록 수동적이긴 하지만, 실생활에서 우리가 스마트폰을 통해 인터넷 서비스에 접속하기 전에 이루어지는 부트스트래핑 과정은 [그림 2]와 같은 일련의 보안 단계를 거친다. 예컨대, ① 스마트폰 USIM 내부에 저장된 사용자 크리덴셜을 통해 인증이 수행되고, ② 인증 결과로부터 생성된 대칭키(Pairwise Transient Key: PTK)로 무선 구간에 대한 암호통신을 수행한다. ③ 스마트폰은 인터넷 사용을 위한 로컬 IP 주소를 AP로부터 할당받고, ④ 사용자



〈자료〉 한국전자통신연구원 자체 작성

[그림 2] WiFi 인터넷 보안 접속 흐름도(IEEE 802.1X)

계정의 과금 정책에 따라 인터넷 서비스를 사용한다. [표 1]은 이러한 IT 인프라의 전형적인 부트스트래핑 과정에서 관찰되는 몇몇 핵심적인 보안 요소들을 정리한 것이다.

사물인터넷(IoT) 보안 얼라이언스는 사물인터넷 제품과 서비스의 보안성 확보를 위해 공통 보안 7원칙을 포함한 “IoT 공통 보안가이드”[3]를 2016년에 발표하였다. [그림 1]의 설치 및 구성보안 단계에서는 이른바 “Secure by Default” 기본 원칙을 통해 보안 설정 과정을 관리자의 취사선택 항목이 아닌 기본 요소로 규정하고 있다. 특히, 보안 프로토콜의 파라미터들을 가장 안전한 값으로 설정하기 위해서 제조사나 서비스 운용사는 부트스트래핑 기술을 이용해서 자동화된 보안 설정 방안을 반드시 제공해야 한다.

[표 1] 부트스트래핑 과정에서의 주요 보안 요소

보안 요소	주요 내용
인증 (authentication)	비밀 크리덴셜 정보를 이용하여 사용자 혹은 기기에 대해 신원(identity)을 검증한다. 대표적인 크리덴셜 종류는 대칭키, 공개키 또는 패스워드 등이 있으며, 이들은 설치 단계에서 기기 내에 주입되거나 이후 보안 설정 단계에서 추가적으로 배포될 수 있다.
인가 (authorization)	신원이 확인된 사용자 또는 기기에게 부여된 권한의 허용 범위를 검증하는 과정이다. 기기가 접근하려는 특정 네트워크 또는 서비스 도메인 등이 대표적인 권한의 대상에 포함되며 이들 권한은 각각 IP 주소 또는 도메인 키 등의 형태로 표현될 수 있다.
계정관리 (accounting)	유료 서비스의 경우 네트워크 자원에 대한 사용 이력 정보를 사용자 또는 기기 계정별로 관리한다. 이러한 이력 정보는 과금뿐만 아니라 비정상 행위 기반의 공격 탐지 시에도 활용될 수 있다.
키분배 (key distribution)	상호 인증된 기기 간 또는 그룹 간의 암호통신을 위해서는 사전에 키 공유가 필요하다. 사전에 미리 배포된 고정키(pre-shared key)보다는 인증 과정에서 크리덴셜 정보를 바탕으로 동적으로 생성된 세션키(transient key) 사용을 권장한다.

〈자료〉 한국전자통신연구원 자체 작성

[그림 1]의 사물인터넷 주기에서 설치 및 보안설정 단계에서는 기기 신원 정보와 더불어 소유권 정보, 비밀키 등의 크리덴셜 정보를 기기에 주입한다. 그리고 기기가 실제로 배속되는 도메인의 관리 서버 접속 정보 및 보안 파라미터 등을 설정하게 된다. 사물인터넷의 특성상, 기기가 검증된 네트워크에 접속하는 과정은 필수적이므로 보안 설정 단계의 역할은 매우 중요하다. 이처럼 부트스트래핑을 설치 단계와 보안 설정 단계로 논리적으로 구분할 수도

[표 2] 대규모 환경에서의 부트스트래핑 솔루션 요구사항

요구사항	주요 내용
R1. 링크계층 독립성(link-layer independent)	LPWAN 등 다양한 사물인터넷 통신 환경에서 운용 가능
R2. 경량화(lightweight protocol)	제약적인 환경에서도 수행 가능한 연산 부하량 최적화
R3. 아이디 연동(identity federation)	이종 도메인 간 아이디 서비스(인증, 인가) 연계 필요
R4. 키관리(flexible key management)	암호통신(신뢰 채널)에 필요한 유연한 키관리 기능 지원
R5. 인증(flexible authentication)	기기의 보안 강도에 부합하는 다양한 인증 방식 지원
R6. 계정관리(accounting)	과금 및 공격행위 감지를 위한 네트워크 사용 이력 관리
R7. 인가(robust authorization)	유연하고 안정적인 권한 관리 메커니즘 제공
R8. 다중 홉 네트워크(multi-hop networks support)	다중 홉으로 구성된 네트워크 환경에서도 운영 가능
R9. 코드 재사용(code reuse)	표준 프로토콜 기반 코드 재사용에 따른 중복 투자 방지
R10. 저전력 장거리 통신 지원(LPWAN support)	초저전력 기기 및 통신 환경에서도 운용 가능
R11. 확장성(scalability)	이종 도메인 및 대규모 기기 환경에서도 운영 가능

〈자료〉 Dan Garcia Carrillo, "A CoAP-Based Bootstrapping Service for Large-Scale Internet-of-Things Networks," University of Murcia, 2018. 07, pp.13-14. 재구성

있지만, 실제 솔루션에서는 반드시 분리하여 구현하지는 않는다. 오히려 부트스트래핑의 구체적인 역할과 요구사항은 사물인터넷 기기가 실제 적용되는 서비스와 통신 환경에 따라 [표 2]처럼 매우 다양하게 제시된다. [표 3]은 주요 부트스트래핑 솔루션의 특성을 보여준다.

앞서 IT 인프라의 스마트폰 인터넷 사용례처럼 충분한 컴퓨팅 자원과 사용자 인터페이스(UI)를 통해 사용자가 네트워크 설정에 직접 개입할 수 있지만, OT 인프라의 사물(기기)로부터 이와 동일한 환경을 기대하기는 어렵다. 특히, 물리적으로 접근이 제한된 OT 인프라에서의 사물인터넷 제품이라면, 기존의 부트스트래핑 기술을 그대로 적용하기는 사실상 불가능하다. 초연결성(hyper-connectivity)으로 특징되는 사물인터넷에서 부트스트래핑 보안 기술은 사람의 개입 없이 기기 스스로 접속 가능한 네트워크를 찾고, 검증과 동시에 접속에 필요한 보안 파라미터를 설정하는 자동화를 추구한다. 이로 인해서 자율성(autonomy)과 신뢰성(trustworthiness)이라는 특징을 갖게 된다. 더불어 대규모 기기 환경에서도 쉽게 운용될 수 있는 확장성(scalability)까지 추가적인 요구사항으로 제시되고 있다[4].

본 고에서는 [표 3]의 부트스트래핑 보안 기술의 대표 솔루션 중에서 데이터 링크계층의 제약 없이(R1) 다중 홉 네트워크 환경(R8)에서도 높은 확장성(R11)을 제공하는 PANA 프로토콜[5]을 설명한다. 이어 사물인터넷 플랫폼의 사실 표준(de facto standard) 가운데 다양한 인증 방식을 통해 차등화된 신뢰성을 제공하는 OCF 온보딩 기술[8]을 두 번째 기술로 소개한다. 마지막으로 유연성과 관련된 표준 기술로서 IETF ANIMA 워킹그룹에서 진행 중인 자율 네트워킹(autonomic networking)의 BRSKI 프레임워크[11]에 대해서 살펴본다.

[표 3] 주요 부트스트래핑 솔루션 특성

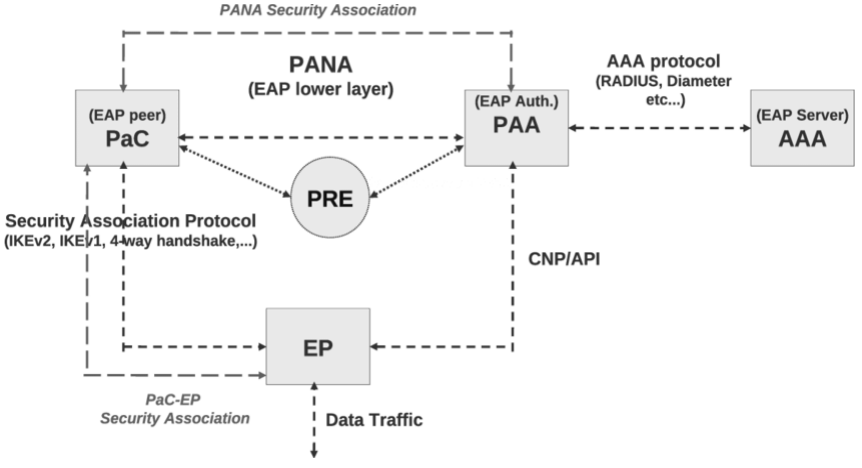
Solution	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11
LoRaWAN with AAA주)	NO	YES	YES	NO	NO	YES	YES	NO	NA	YES	YES
IEEE 802.15.9	NO	YES	YES	YES	YES	YES	YES	YES	NA	NO	YES
DTLS	YES	NO	NO	NO	YES	NO	NO	NO	YES	NO	NO
HIP-DEX	YES	YES	NO	YES	NO	NO	NO	NO	NO	NO	NO
EST	YES	NO	YES	YES	NO	NO	NO	NO	NO	NO	YES
PANA	YES	NO	YES	YES	YES	YES	YES	YES	NO	NO	YES
CoAP	YES	YES	NO	NO	NO	NO	NO	YES	YES	NO	NO

주) Authentication Authorization Accounting
〈자료〉 Dan Garcia Carrillo, “A CoAP-Based Bootstrapping Service for Large-Scale Internet-of-Things Networks,” University of Murcia, 2018. 7, p.17. 재구성

III. 부트스트래핑 보안 기술 표준화 현황

1. PANA

현재 사물인터넷 환경에서 대표적인 부트스트래핑 기술로 사용 중인 PANA(Protocol for Carrying Authentication for Network Access)는 ZigBee IP[6]와 IEEE 802.15.9 표준 규격[7]에서 각각 네트워크 인증과 키관리 프로토콜(Key Management Protocol: KMP)로 지정되었다. PANA는 앞서 스마트폰 인터넷 사용례에서 적용된 IEEE 802.1X와 유사하게 네트워크 인증과 데이터 링크 계층의 암호통신에 필요한 크리덴셜 정보(Link Layer Credentials: LLC) 공유를 위해 사용된다. 하지만 이종의 대규모 사물인터넷 통신 환경에서 요구되는 링크계층 독립성을 제공한다는 점에서 차별성을 보인다. 이름처럼, PANA 프로토콜은 인증 프레임워크 프로토콜인 EAP(Extensible Authentication Protocol)를 운반하는 캡슐 역할을 수행한다. 실제로 신원 검증은 EAP 내부 인증 메커니즘을 통해 이루어지기 때문에 EAP가 지원하는 다양한 인증 방식을 네트워크 접근제어에 활용하게 된다. 더불어 UDP 프로토콜을 사용하기 때문에 특정 데이터 링크 계층에 의존하지 않고 응용 내에서 다루어질 수 있다.

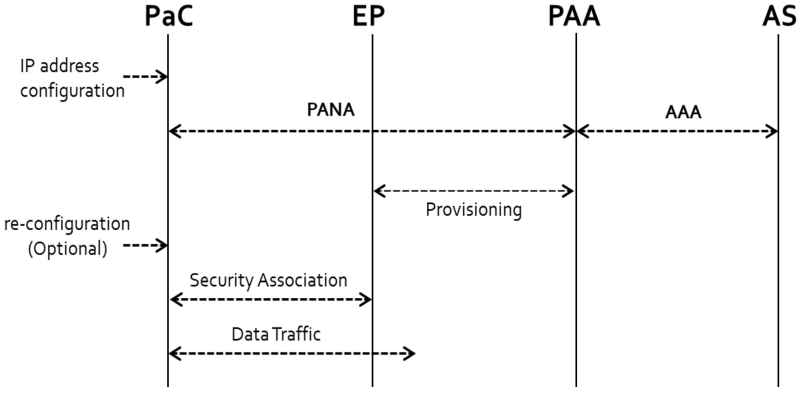


〈자료〉 Pedro Moreno Sanchez, "PANATIKI: A Network Access Control Implementation Based on PANA for IoT Devices," Sensors, 2013. 11, p.14893.

[그림 3] PANA 프로토콜 구조

PANA는 [그림 3]과 같이 크게 5개의 논리적인 네트워크 요소로 구성된다. PANA Client (PaC)는 네트워크 인증과 비밀키 공급을 요구하는 신청자(EAP peer)로서 사물인터넷 단말기에 동작한다. 상대적으로 PANA Agent(PAA)는 PANA Client 인증 요청을 검증하는 인증자(EAP authenticator)로서 코디네이터 또는 AP 같은 게이트웨이 장비에서 운영된다. 앞서 설명된 바와 같이, 기기(PaC) 인증에 필요한 크리덴셜 검증은 실제로 AAA 서버에서 이루어진다. 그리고 PAA와 AAA 서버 간 통신은 RADIUS, Diameter 등의 AAA 전용 프로토콜을 사용한다. 현재 다양한 종류의 인증 방식을 제공하지만, 보안성이 높고 개방형 구조를 선호하는 사물인터넷 환경에서는 인증서 기반의 EAP-TLS 사용을 권장한다[6],[8].

[그림 4]는 PANA의 동작 과정을 보여준다. 네트워크 경계(perimeter) 지점에 위치한 Enforcement Point(EP)는 PAA의 직접적인 통제 속에 Authentication Server(AS)의 인증 결과에 따라 기기(PaC)에서 네트워크로 유입되는 데이터 트래픽을 제어한다. 논리적으로 PAA와 EP는 구분된 기능이지만 실제로 한 장비 안에서 통합되어 구현될 수 있다. 마지막 요소인 PANA Relay(PRE)는 PaC와 PAA 간 직접 통신이 어려운 멀티홉 환경에서도 메시지를 교환할 수 있도록 중계 서비스를 담당한다. 따라서 기기의 통신 환경에 따라 선택적으로 구성될 수 있다. PANA가 지닌 높은 활용성은 다양한 통신 환경을 지원할 뿐만 아니라 기기의 신원을 검증하는 인증 기능, 그리고 신뢰채널 기능인 보안 연관(Security Association: SA)으로부터 기인된다. 기기와 PAA(PaC-PAA) 또는 기기와 EP(PaC-EP) 간에는 보안 채널인 SA를 통해 데이터링크 계층에서의 암호통신을 지원할 수 있다.

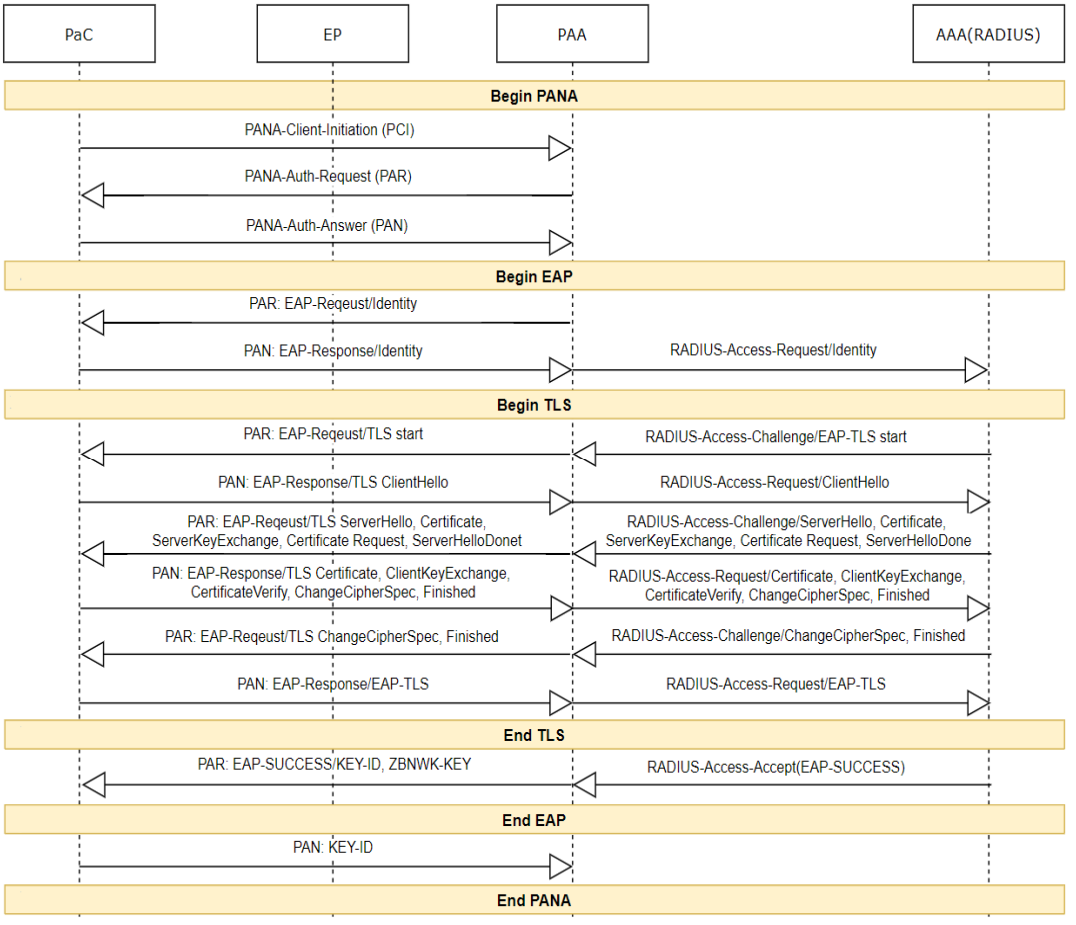


〈자료〉 “Protocol for Carrying Authentication for Network Access(PANA) Framework,” RFC 5193, 2008. 5.

[그림 4] PANA 시그널링 흐름도

EAP 인증이 성공적으로 끝나면 AAA 서버는 EAP 최종결과(Access-Accept)와 더불어 비밀 크리덴셜 정보(Master Session Key: MSK)를 PAA에게 전달한다. 이후 PaC와 PAA는 MSK로부터 유도된 비밀키를 공유하게 되고 신뢰 채널인 SA를 통해 양방향 인증과 암호통신을 수행하게 된다. 추가적으로 ZigBee IP 규격에서는 PANA의 신뢰 채널인 SA를 통해 네트워크 키 정보(ZBNW-KEY and KEY-ID)를 분배하거나 갱신할 수 있다[6].

[그림 5]는 ECC 인증서 기반의 TLS를 인증 수단으로 사용하고 RADIUS AAA 프로토콜을 적용한 ZigBee IP 규격의 적용 사례를 보여준다. 프로토콜이 진행함에 따라 PANA, EAP, TLS 순서대로 시작되고 반대 순서로 종료되는 과정을 확인할 수 있다. 인증서 기반의 검증



〈자료〉 한국전자통신연구원 자체 작성

[그림 5] ZigBee IP PANA 프로토콜 동작 흐름도

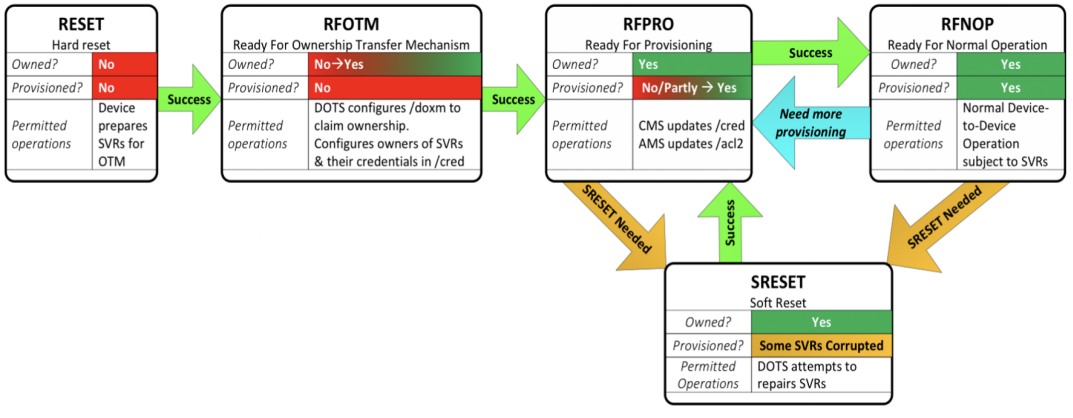
절차가 성공적으로 끝나면 AAA 서버는 결과(Access-Accept)와 함께 MSK를 RADIUS 프로토콜을 통해 PAA에게 전달한다. PaC와 PAA는 EAP-TLS의 성공적인 인증 결과와 MSK로부터 상호 인증과 암호통신에 필요한 비밀키 정보를 공유하게 된다.

2. OCF 온보딩

OCF 보안 구조의 목표는 사물인터넷 생태계에서 OCF 자원(resource)으로 표현되는 모든 기기들의 상태나 제어 명령어를 외부의 위협으로부터 보호하는 데 있다. 이때 보호 대상은 인증 가능한(certifiable) 논리적 개체(logical entity)로 표현되며, 보안 정책과 접근제어 메커니즘을 통해서 OCF 보안 도메인 소유자(security domain owner) 또는 관리자는 보안 기능을 동작시키게 된다.

OCF 보안 모델에서 보안 정책과 접근제어는 기기의 신원과 역할을 표현하는 크리덴셜을 기반으로 이루어진다. 이러한 크리덴셜 정보는 키 구조에 따라 대칭 또는 비대칭 구조를 가지는 가상의 보안 자원인 SVR(secure virtual resource) 형태로 표현된다. 따라서 모든 OCF 기기는 서비스를 시작하기 이전에 비밀키 또는 공개키 형태의 크리덴셜을 보유하게 된다. 현재 OCF 보안 규격서[8]는 크리덴셜 종류로 일대일 대칭키, 그룹키, 인증서, 공개키의 다양한 형태를 정의하고 있다.

OCF 보안에서의 부트스트래핑은 기능적으로 온보딩과 프로비저닝으로 구분되며 각각 기

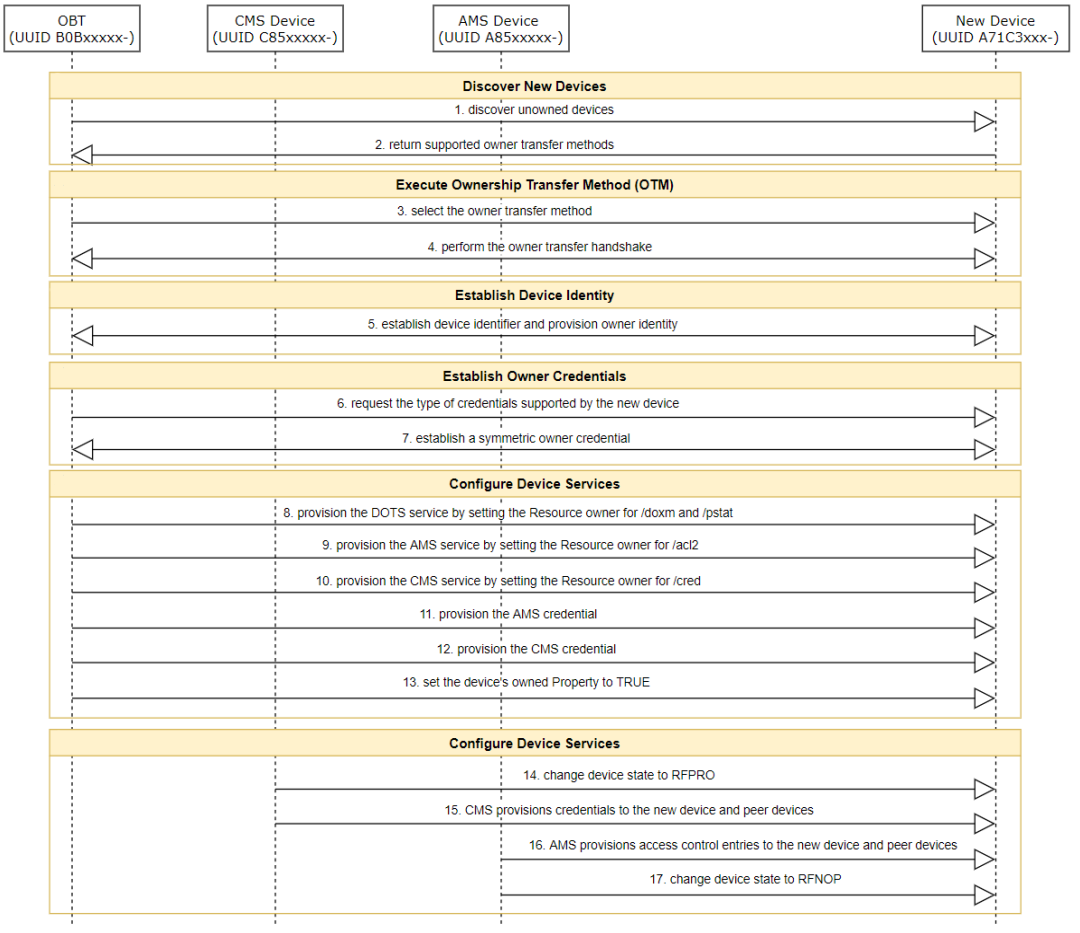


〈자료〉 Brian A. Scriber, “Changing the World: IoT Chaos as a Ladder to Improving Security,” NCTA, 2019. 10, p.7.

[그림 6] OCF 부트스트래핑 상태 전이도

기 소유권 설정과 보안 크리덴셜의 설치 여부에 따라 [그림 6]의 단계로 정의된다. 최초 설치 단계인 온보딩 과정에서는 기기에 대한 소유권을 설정하고, 프로비저닝 단계에서는 기기 신원 및 접근권한 정보를 기기에 공급한다. 이러한 두 단계를 성공적으로 마친 뒤에 비로소 다른 기기와의 통신 및 서비스 연계가 가능해진다. [그림 6]과 같이 OCF 기기는 초기 설치 단계인 RESET(Hard Reset) 상태부터 부트스트래핑이 완료되고 정상 서비스가 가능한 RFNOP(Ready For Normal Operation) 상태에 이르기까지 소유권과 크리덴셜 설치 여부에 따라 상태 전이가 발생된다.

기기 소유권을 설정하는 온보딩 기술은 소유자로 하여금 해당 기기를 자신의 보안 도메인,



〈자료〉 한국전자통신연구원 자체 작성

[그림 7] OCF 부트스트래핑 흐름도

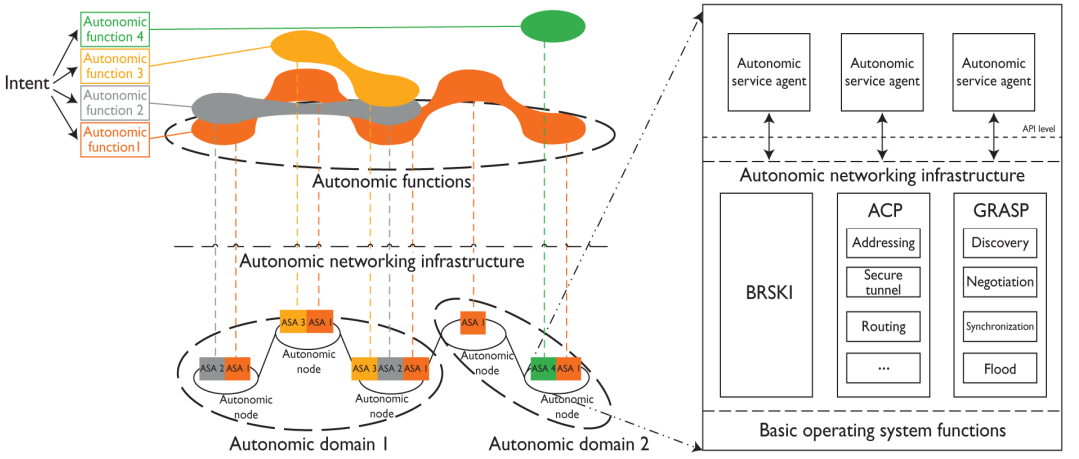
즉 소유자가 통제하는 개인 네트워크로 편입시키는 과정이다. 따라서 OCF 보안 영역에 대한 소유자 혹은 기기 사용자는 온보딩 기술을 통해 미등록된 기기에 대한 통제권을 확보하게 되고 이미 등록된 다른 기기와 사용자 중심의 전용 네트워크를 구축하게 된다. 현재 OCF 온보딩 소유권 설정 서비스(Device Ownership Transfer Service)에서 사용 가능한 인증 방식으로 단순 페어링(just-works), 핀입력(random PIN), 제조사 인증서(manufacturer certificate) 등이 있다. 이 가운데 제조사 인증서 기반의 인증 방법을 가장 권장하지만, 검증에 필요한 PKI(Public Key Infrastructure) 시스템 구축이 추가적으로 요구된다.

프로비저닝은 기기 인증 및 소유권 설정이 완료된 이후에 진행되는 보안 설정 단계로서 기기 식별자, 소유자 정보, 기타 보안 속성을 표현하는 크리덴셜 정보를 기기에 추가로 공급한다. 현재 기기 인증과 암호통신은 DTLS(Datagram Transmission Layer Security) 프로토콜을 활용한다. 위의 모든 과정을 정상적으로 마친 이후에, 서비스 동작이 가능한 준비 상태(RFNOP)로 전이한다. [그림 7]은 OCF 부트스트래핑의 전체 과정을 나타낸다.

3. BRSKI

자율 네트워킹은 망의 제어 및 관리를 네트워크 관리자의 수동적인 설정이 아닌, 인프라의 지능적 자가관리(self-management)를 통해 사람의 개입을 최소화하거나 자율적으로 운영하는 새로운 패러다임이다[9]. [그림 8]은 자율 네트워킹의 참조모델을 나타낸다. IETF에서는 ANIMA(Autonomic Networking Integrated Model and Approach) 워킹그룹 내에서 자율 네트워킹을 위한 표준화를 진행한다[10]. 그 가운데 BRSKI(Bootstrapping Remote Secure Key Infrastructures) 프로토콜[11]은 외부의 악의적인 공격으로부터 자율 네트워킹 인프라를 보호하고, 자가보호(self-protection) 속성을 실현하는 임무를 맡는다. 인프라에 등록될 기기가 물리적 접근이 제한된 서비스 현장에 배치되더라도, 제로 터치 부트스트래핑 기술(Secure Zero Touch Provisioning: SZTP)[12]인 BRSKI를 통해서 신원 정보 검증과 네트워크 설정을 자동으로 처리하게 된다.

유사한 목적으로 EST(Enrollment over Secure Transport)[13] 프로토콜이 대표적인 부트스트래핑의 자동화 기법으로 언급되지만, 인증서 검증에 필요한 대칭키 또는 트러스트 앵커(Trust Anchor) 정보를 기기에 미리 탑재해야 하는 한계를 지닌다. 반면, SZTP는 과정 자체의 자동화보다 신뢰 사슬(Chain of Trust)을 확장해 나가는 데 필요한 신뢰 구축 과정

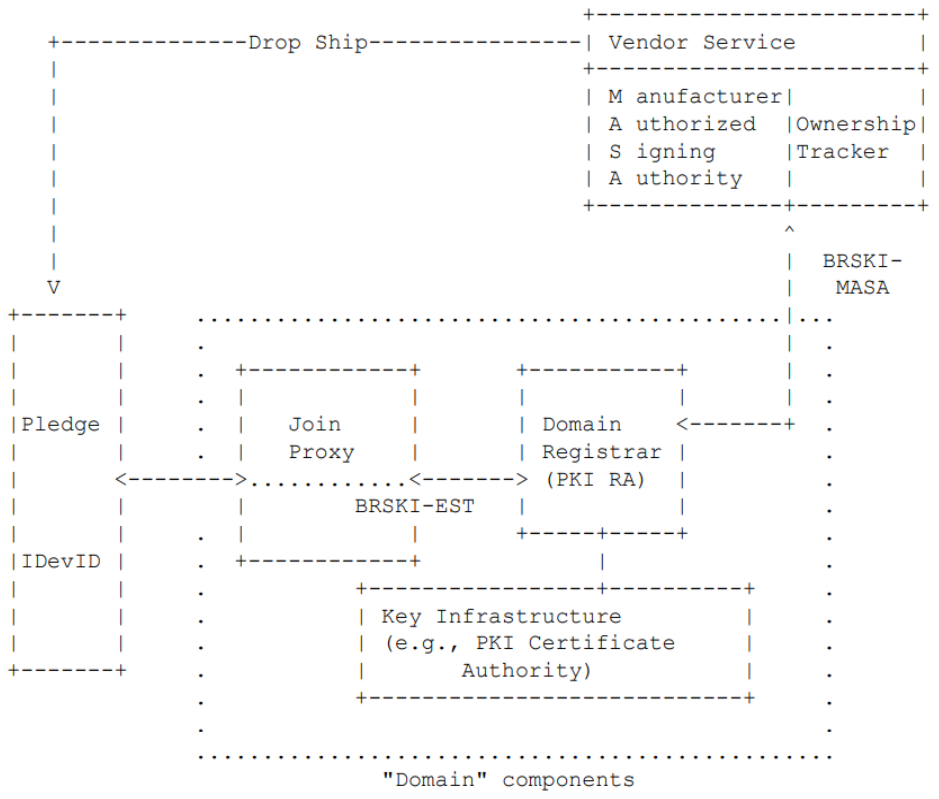


〈자료〉 Xinjian Long, "Autonomic Networking: Architecture Design and Standardization," IEEE Internet Computing, 2017. 1.

[그림 8] 자율 네트워킹 참조모델

의 유연성에 역점을 두고 있다. 따라서 도메인에 대한 사전 정보가 전혀 없는 신규 기기가 이미 검증된 네트워크에 접속하고 새로운 서비스 도메인에 참여하기 위해서는 BRSKI와 같은 제로 터치 프로비저닝 기술이 필요하다. [그림 8]의 참조모델에서 소유권 정보를 사전에 탑재하지 않은 신규 기기를 보안상으로 적대적이고 물리적으로 접근이 제한적인 환경에 배속하려는 경우, 부트스트래핑 과정을 통해서 해당 기기는 자율 네트워킹 인프라에 편입된다.

BRSKI 프로토콜은 [그림 9]와 같이 신규 등록 대상 기기인 Pledge, 기기 제조 및 소유권 상태 정보를 제공하는 제조사의 Vendor Service, 신규 기기의 네트워크 접속 관리 및 도메인 등록 과정을 주관하는 운용사의 Domain Registrar로 이루어진다. 제조사는 기기의 초기 신원 정보인 기기 인증서(IDevID)를 기기(Pledge)에 주입하고 MASA(Manufacturer Authorized Signing Authority) 서비스를 통해 기기의 신원 정보 검증 및 소유권 상태에 대한 추적 서비스를 외부에 제공한다. 기기의 실제 소유자인 서비스 운용사는 도메인 관리 시스템인 Domain Registrar를 통해 신규로 배치된 기기를 검증하고 서비스 도메인 내부에 등록한다. 설치 위치 및 통신 환경에 따라 중계 서비스를 위한 프록시(Join Proxy), 또는 인증서 사용에 따라 키 인프라(Key Infrastructure)를 추가적인 시스템 요소로 구성한다. 검증 절차를 수행할 때 [그림 9]의 BRSKI 연결 구조에서 주목할 부분은, 모든 메시지가 서비스 운용사의 Domain Registrar를 반드시 경유하여 제조사의 MASA에 도달한다는 점

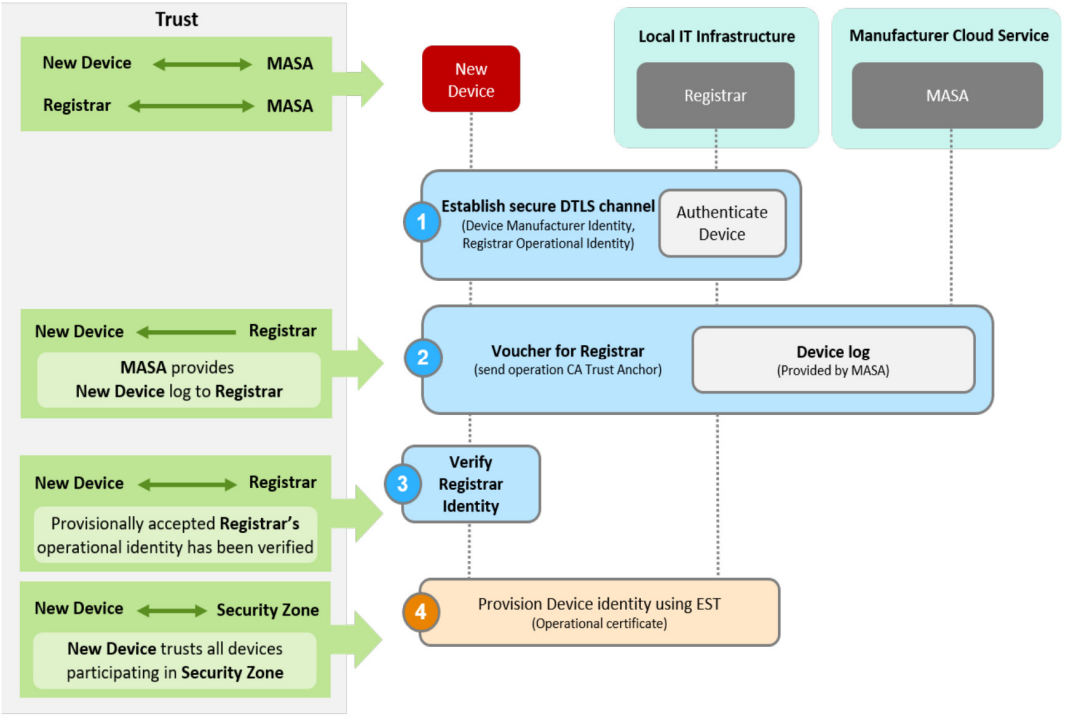


〈자료〉 “Bootstrapping Remote Secure Key Infrastructure(BRSKI),” RFC 8995, 2021. 5.

[그림 9] BRSKI 구성 요소 및 연결 구조

이다. 이는 제조사로부터 프라이버시를 보호하기 위해 기기 소유자인 서비스 운용사가 모든 메시지에 대한 중계권을 독점적으로 확보하려는 설계상의 의도로 파악된다.

[그림 10]은 BRSKI를 활용하여 사물인터넷 빌딩 서비스에 적용한 사례[14]를 보여준다. 초기 신원 정보(IDevID)를 탑재한 신규 기기는 인프라 관리 시스템(Domain Registrar)을 거쳐 제조사 서비스(MASA)에게 바우처(Voucher) 발급을 요청한다. 발급된 바우처를 통해 인프라(Domain Registrar)의 신원 및 소유 권한 정보를 검증한다. 이렇게 바우처를 통해 우회적으로 검증하는 모델이 BRSKI의 대표적인 특징이 된다. 성공적으로 바우처 검증을 마친 신규 기기(Pledge)는 인프라 관리 시스템(Domain Registrar)과의 상호 신뢰 관계를 구축한다. 기기는 새로 구축된 상호 신뢰 관계를 바탕으로 인프라 전용의 신규 신원 정보를 발급받고 인프라의 구성원으로서 다른 기기와 통신이 가능해진다. 따라서 신뢰 관계 관점에



〈자료〉 Piotr Polak, "Security Architecture for the Internet of Things(IoT) in Commercial Buildings," 2018. 3, p.10.

[그림 10] BRSKI 기반 기기 신원 검증 절차

서 신규 기기는 최종적으로 인프라 내부의 다른 기기와 연결됨을 확인할 수 있다.

IV. 결론 및 시사점

부트스트래핑은 기기가 서비스를 시작하는데 필요한 일련의 준비과정으로 보안 관점에서는 외부와의 신뢰 관계를 구축하는 출발점이 된다. 최근 기기 및 통신방식의 다양성과 이질성으로 특징되는 사물인터넷 환경에서 신뢰 관계도 다양하고 복잡해짐에 따라 부트스트래핑의 범위와 역할은 점점 확대되는 추세이다. 본 고에서는 사물인터넷 기기를 위한 부트스트래핑 보안 기술로서 PANA, OCF 온보딩, BRSKI 표준 프로토콜을 소개하였다. 각각 확장성, 신뢰성, 유연성이라는 측면에서 대표적인 기술로 소개하였으나 최근에는 더 나아가 자율성 및 지능성을 향상시키는 방향으로 부트스트래핑 보안 기술 개발이 요구된다.

전통적인 IT 인프라에서 부트스트래핑 보안 기술[15]은 하드웨어로 구성된 신뢰의 밑바탕(Root of Trust)을 서버 또는 데이터로 확장하는 데 주력하고 있다. 반면, 사물인터넷에서 부트스트래핑 보안 기술은 신뢰 관계가 잘 정립된 IT 인프라의 신뢰 사슬을 OT 인프라의 기기로 확장하는 연결 고리 역할을 수행한다. 이런 관점에서 향후 부트스트래핑 기술은 자율적이고 지능적으로 진화하는 사물인터넷과 사이버 물리 시스템(Cyber Physical System: CPS)에서 "Secure By Default" 기본 원칙을 실제로 준수하고 수행하는 주체가 된다.

참고문헌

[1] Dhirani LL, "Industrial IoT, Cyber Threats, and Standards Landscape: Evaluation and Roadmap," Sensors, 2021, 21(11):3901, https://doi.org/10.3390/s21113901

[2] M. Sethi, "Secure IoT Bootstrapping: A Survey", 국제인터넷표준화기구(IETF), 2021. 4.

[3] IoT 보안얼라이언스, "IoT 공통 보안 원칙 v1.0", 미래창조과학부, 2016. 6.

[4] D. Garcia, "A CoAP-Based Bootstrapping Service for Large-Scale Internet-of-Things Networks," Ph. D. dissertation, University of Murica, 2018. 7.

[5] P. Jayaraman, "Protocol for Carrying Authentication for Network Access(PANA) Framework," RFC 5193, 2008. 5.

[6] ZigBee Alliance, "ZigBee IP Specification," 2013. 2.

[7] IEEE, "IEEE Recommended Practice for Transport of Key Management Protocol(KMP) Datagrams," IEEE Standard 802.15.9, 2016. 8.

[8] Open Connectivity Foundation, "OCF Security Specification VERSION 2.2.3," 2021. 4.

[9] 신승재, 윤승현, 이범철, 김상기, "자율네트워킹 연구동향", 한국전자통신연구원, 전자통신동향분석, 32권 1호, 2017. 2, pp.25-34.

[10] T. Eckert, "An Autonomic Control Plane(ACP)," RFC8994, 2021. 5.

[11] M. Pritikin, "Bootstrapping Remote Secure Key Infrastructure(BRSKI)," RFC 8995, 2021. 5.

[12] K. Watsen, "Secure Zero Touch Provisioning(SZTP)," RFC 8572, 2019. 4.

[13] M. Pritikin, "Enrollment over Secure Transport," IETF 7030, 2013. 10.

[14] Piotr Polak, "Security Architecture for the Internet of Things(IoT) in Commercial Buildings," Fairhair Alliance, 2018. 3.

[15] 김지우, 이상길, 고재용, 이철훈, "TPM을 활용한 임베디드 시스템 환경의 보안 부팅 구현", 정보보호학회논문지, 29권 5호, 2019. 10, pp.949-960.