

OCF Core Specification Extension

Wi-Fi Easy Setup

VERSION 1.3.1



OPEN CONNECTIVITY
FOUNDATION™

CONTACT admin@openconnectivity.org
Copyright Open Connectivity Foundation, Inc.
© 2016- 2018. All Rights Reserved.

법적 고지 사항

이 문서에 기재된 내용 중 그 어느 것도 명시적 또는 암시적으로 기재 내용에 있어서 어떠한 형태의 사용 허가를 부여하거나 이 문서의 작성자 또는 개발자 중 어느 누구도 소유 또는 관할하는 어떠한 지식재산에 대해 어떠한 형태의 사용 허가도 부여하는 것을 의미하지 않습니다. 여기에 포함된 정보는 "있는 그대로" 제공되며, 적용 가능한 법에 의해 허용되는 최대 한도까지 이 시방서의 작성자 및 개발자는 특정한 목적을 위한 판매 적격성 또는 적합성의 암시적 보증을 포함하지만 이에 한정되지 않는 명시적 또는 암시적인 성문법 또는 불문법 상의 기타 모든 보증 및 조건에 대해 일절 책임을 지지 않습니다. OPEN CONNECTIVITY FOUNDATION, INC.는 비침해, 정확성, 또는 바이러스 비 감염에 대한 모든 보증에 대해서도 일절 책임을 지지 않습니다.

OCF 로고는 미국 및 다른 국가에서 Open Connectivity Foundation, Inc 의 상표입니다. *그 밖의 명칭 및 상표는 해당하는 소유자의 자산일 수 있습니다.

Copyright © 2016-2018 Open Connectivity Foundation, Inc. All rights reserved.

이들 저작물의 복사 또는 기타 형태의 복제 및/또는 배포는 엄격하게 금지되어 있습니다.

Do not translate this text: For Translation to Local Language

Translate the below into Korea:

- 이 OCF 시방서의 번역본은 OCF 기반의 제품 개발을 장려하고 이에 도움이 되도록 영문 원본 으로부터 작성되었습니다. 영문 시방서의 정확한 번역을 위한 모든 노력을 기울이기는 하였지만 이 번역본을 규정으로 간주해서는 안 됩니다. OCF 인증 프로그램은 명백하게 영문 시방서를 기준으로 개발되어야 하며, 어떠한 면제 또는 면책 요구도 영문 시방서의 문구를 기준으로 평가되어야 합니다.
- 최신 영문판 시방서의 공개로부터 번역본의 공개까지는 소정의 지연이 있을 수 있습니다.
- OCF 시방서의 최신 영문판 및 해당 번역본에 관해서는 <https://openconnectivity.org/developer/specifications>를 참조하여 주십시오.

목차

18			
19			
20	1	적용 범위	8
21	2	인용 표준	8
22	3	용어, 정의, 기호 및 약어	9
23	3.1	용어와 정의.....	9
24	3.2	협약	9
25	3.3	Data type	9
26	4	문서 규약 및 구성	10
27	5	개관.....	11
28	5.1	개요	11
29	5.2	아키텍처	11
30	5.3	시나리오 예.....	11
31	6	Resource model.....	12
32	6.1	개요	12
33	6.2	EasySetup Resource	12
34	6.2.1	개략.....	12
35	6.2.2	Resource	12
36	6.3	WiFiConf Resource Type	14
37	6.3.1	개요.....	14
38	6.3.2	Resource Type	14
39	6.4	DevConf Resource Type	15
40	6.4.1	Introduction	15
41	6.4.2	Resource Type	16
42	7	네트워크 및 연결성	17
43	8	기능적 상호 작용	17
44	8.1	온보딩, 프로비저닝, 및 구성	17
45	8.2	Resource 탐색	17
46	8.3	Easy Setup Resource의 검색 및 업데이트.....	17
47	8.4	에러 처리	18

48	8.5	Easy Setup Flow 예	18
49	9	보안.....	20
50	Annex A (규정)	Resource Type 정의.....	21
51	A.1	Resource Type 정의 목록.....	21
52	A.2	Easy Setup Collection Baseline Interface	21
53	A.2.1	개요.....	21
54	A.2.2	URI 예	21
55	A.2.3	Resource Type	21
56	A.2.4	RAML 정의	21
57	A.2.5	Property 정의	25
58	A.2.6	CRUDN 동작.....	25
59	A.3	Wi-Fi Configuration Resource Baseline Interface	25
60	A.3.1	개요.....	25
61	A.3.2	URI 예	26
62	A.3.3	Resource Type	26
63	A.3.4	RAML 정의	26
64	A.3.5	Property 정의.....	32
65	A.3.6	CRUDN 동작.....	32
66	A.4	Device Configuration	32
67	A.4.1	개요.....	32
68	A.4.2	URI 예	33
69	A.4.3	Resource Type	33
70	A.4.4	RAML 정의	33
71	A.4.5	Property Definition	35
72	A.4.6	CRUDN behavior.....	35
73	Annex B (참고)	Swagger2.0 정의	36
74	B.1	Device 구성	36
75	B.1.1	개요.....	36
76	B.1.2	잘 알려진 URI	36
77	B.1.3	Resource Type	36
78	B.1.4	Swagger2.0 정의	36

79	B.1.5	Property Definition	39
80	B.1.6	CRUDN behaviour	39
81	B.2	Easy Setup Collection.....	39
82	B.2.1	개요.....	39
83	B.2.2	잘 알려진 URI	39
84	B.2.3	Resource Type	39
85	B.2.4	Swagger2.0 정의	39
86	B.2.5	Property Definition	56
87	B.2.6	CRUDN behaviour.....	58
88	B.3	Wi-Fi Configuration Resource	59
89	B.3.1	개요.....	59
90	B.3.2	잘 알려진 URI	59
91	B.3.3	Resource Type	59
92	B.3.4	Swagger2.0 정의	59
93	B.3.5	Property 정의	66
94	B.3.6	CRUDN 동작.....	67
95			
96			

그림 목차

97	
98	그림 1. Easy Setup 구현 아키텍처 11
99	그림 2. Easy Setup Resource Type 12
100	그림 3. Easy Setup Flow (참고) 19
101	
102	

103
104
105
106
107
108
109
110
111
112
113

표

표 1. EasySetup Resource Type	12
표 2. "oic.r.easysetup" Resource Type 정의	13
표 3. WiFiConf Resource Type	14
표 4. "oic.r.wificonf" Resource Type 정의	15
표 5. DevConf Resource Type	16
표 6. "oic.r.devconf" Resource Type 정의	16
표 7. 알파벳순 resource 목록	21

1 적용범위

이 표준은 Wi-Fi Easy Setup 의 요건을 충족시키기 위해 OCF Core 시방서에 정의된 기능에 대한 기능적 확장을 정의한다. 이 시방서는 기능성을 조장하는 새로운 Resource Type 및 OCF Core 시방서에 정의된 기존 기능에 대한 모든 확장을 규정한다.

2 인용표준

다음의 인용표준은 전체 또는 부분적으로 이 표준의 적용을 위해 필수적이다. 발행연도가 표기된 인용표준은 인용된 판만을 적용한다. 발행연도가 표기되지 않은 인용표준은 최신판(모든 추록을 포함)을 적용한다.

OCF Core Specification, *Open Connectivity Foundation Core Specification*, Version 1.3

Available at: https://openconnectivity.org/specs/OCF_Core_Specification_v1.3.0.pdf

Latest version available at: https://openconnectivity.org/specs/OCF_Core_Specification.pdf

OCF Security Specification, *Open Connectivity Foundation Security Capabilities*, Version 1.3

Available at: https://openconnectivity.org/specs/OCF_Security_Specification_v1.3.0.pdf

Latest version available at: https://openconnectivity.org/specs/OCF_Security_Specification.pdf

IEEE 802.11:2016, IEEE Standard for Information technology—Telecommunications and information exchange between systems Local and metropolitan area networks—Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, December 2016

<https://standards.ieee.org/findstds/standard/802.11-2016.html>

IETF RFC 7159, *The JavaScript Object Notation (JSON) Data Interchange Format*, March 2014

<https://www.rfc-editor.org/info/rfc7159>

IETF RFC 7252, *The Constrained Application Protocol (CoAP)*, June 2014

<https://www.rfc-editor.org/info/rfc7252>

JSON Schema Validation, *JSON Schema: interactive and non-interactive validation*, January 2013

<http://json-schema.org/latest/json-schema-validation.html>

OpenAPI specification, *aka Swagger RESTful API Documentation Specification*, Version 2.0

<https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md>

3 용어, 정의, 기호 및 약어

이 표준의 목적을 위하여 용어와 정의는 OCF Core 시방서에서 주어지고 다음을 적용한다.

3.1 용어와 정의

OCF Core 시방서에 정의된 것에 다음을 추가한다.

3.1.1

Easy Setup 등록

Easy Setup 등록은 Easy Setup 중에 Enrollee 가 Mediator 를 통해 Easy Setup Resource 에 액세스함으로써 Enroller 의 정보를 구성하는 단계이다.

3.1.2

Enrollee

구성 및 접속할 Device. 예를 들어, 에어컨, 프린터.

3.1.3

Enroller

Enrollee 가 접속하는 대상 네트워크 엔티티. 예를 들어, Wi-Fi AP.

3.1.4

Mediator

Enrollee 가 대상 네트워크(Enroller)에 접속할 수 있도록 하는 Device. Mediator 는 Enrollee 에게 구성 정보를 전달한다. 예를 들어, 휴대폰.

3.1.5

Easy Setup

Mediator 를 사용하여 (Enrollee 에게 필수 정보를 전달함으로써) Enrollee 를 구성하는 과정.

3.1.6

Soft AP

전용 Access Point 와 달리 device 에 호스트되는 Software Enabled Access Point.

3.2 협약

이 시방서에서 다수의 용어, 조건, 메커니즘, 시퀀스, 파라미터, 이벤트, 상태, 또는 유사한 용어는 각 단어의 첫 번째 문자를 대문자로 표기하고 나머지는 소문자로 표기한다 (예: Network Architecture). 이러한 단어가 소문자로 표기되었을 때는 일반적인 기술적 영어의 의미를 갖는다.

3.3 Data type

OCF Core 시방서의 정의 참조.

4 문서 규약 및 구성

이 시방서에서 기능은 다음과 같이 필수(Required), 권고(Recommended), 허가(Allowed), 또는 기피(DEPRECATED)로 분류된다.

필수 (강제 또는 의무적) (M)

- 이러한 기본 기능은 OCF Resource Type 시방서를 준수하도록 구현되어야 한다. “하지 말아야 한다”나 “금지한다” 등의 구절은 금지되는, 즉 수행하는 경우 구현이 시방서를 준수하지 않음을 의미하는 행위를 나타낸다.

권고 (또는 제안) (S)

- 이러한 기능은 Core Architecture 에 의해 지원되는 기능을 부가하며, 구현되는 것이 좋다. 권고 기능은 통상적으로 중대한 복잡성의 증가 없이 OCF Core Architecture 의 기능을 이용한다. 규정 준수 테스트를 위해 권고 기능이 구현된다면 이 가이드라인에 따르는 특정 요건을 만족해야 한다. 일부 권고 기능은 추후에 필수 요건이 될 수 있다. “하지 않는 것이 좋다”라는 표현은 허용되지만 권고하지 않는 작용을 나타낸다.

허가 (또는 허용) (O)

- 이러한 기능은 Core Architecture 에 의해 필수적이지도 않을 뿐더러 권고되지도 않지만, 기능을 구현하는 경우에는 이들 가이드라인에 따르는 특정 요건을 만족해야 한다.

기피

- 이에 해당하는 기능은 이 시방서에서 설명은 하고 있지만 역 호환성을 제외하고는 구현되어서는 안된다. 현재 시방서에 따르는 동작 동안 기피된 기능의 발생은 구현 동작에 어떤 영향도 끼치지 않으며 어떠한 에러 상태도 생성하지 않는다. 역 호환성은 특정된 대로 특징이 구현되고 기능할 것을 요구할 수 있지만, 이 시방서에 따르는 구현에 의해 사용되어서는 안된다.

조건부 허용 (CA)

- 정의 또는 행위가 조건에 의존한다. 특정 조건이 만족되면 정의 또는 행위가 허용되고 그렇지 않으면 허용되지 않는다.

조건부 필수 (CR)

- 정의 또는 행위가 조건에 의존한다. 특정 조건이 만족되면 정의 또는 행위가 필수로 된다. 그렇지 않으면, 특별히 허용되지 않는다는 기재가 없는 한 디폴트로 허용된다.

문자 그대로 해석되는 String 은 “인용부호”를 사용한다.

강조하는 단어는 *이탤릭체*로 표기한다.

5 개관

5.1 개요

이 시방서는 이미 구성된 OCF Device 또는 온보딩 툴을 사용하여 새로운 OCF Device 를 설정 및 구성하는 방법을 설명한다.

설명된 설정 및 구성 메커니즘은 선택적이며 그 밖의 메커니즘의 사용도 허용된다.

구체적으로 여기서 설명하는 방법은 신규 Device 로 다음을 포함하는 필수 정보를 전달하는 것을 허용한다.

- 로컬 네트워크 연결 정보. 예를 들어, Wi-Fi 의 경우 Wi-Fi 액세스 포인트 정보.
- Device 구성: 추가 Device 구성 정보.

Easy Setup 은 추후 그 밖의 적절한 기술을 통합함으로써 개선될 수 있다.

5.2 아키텍처

그림 1 은 Easy Setup 을 구현하는 아키텍처적인 접근을 보여준다.

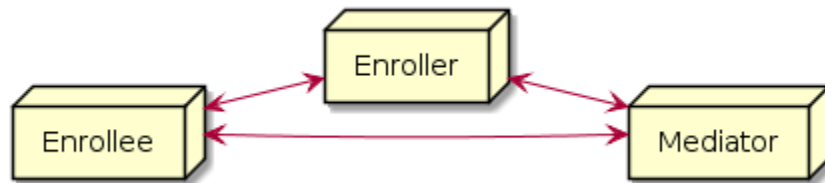


그림 1. Easy Setup 구현 아키텍처

Easy Setup 은 다음과 같은 role 을 정의한다: Enrollee, Enroller, 및 Mediator. 이들 role 에 대한 정의는 섹션 3.1 을 참조하기 바란다.

5.3 시나리오 예

다음의 시나리오는 일반적인 설정 사례를 제시한다.

구성 정보 및 수행되는 단계는 Device 의 유형 및 상태에 따라 달라질 수 있다.

1. Enrollee 가 Easy Setup 모드로 전환된다 (Device 를 처음으로 개봉하면 기본적으로 이 모드로 되어 있을 수 있다).
2. Mediator 가 Enrollee 를 탐색하여 접속한다.
3. Mediator 가 Enrollee 의 보안 프로비저닝을 수행한다.
4. Mediator 가 Enrollee 에게 Wi-Fi 설정 정보를 전달한다.

226 5. Mediator로부터 수신한 정보를 사용하여 Enrollee가 Enroller (Wi-Fi AP)에 접속한다.

227 **6 Resource model**

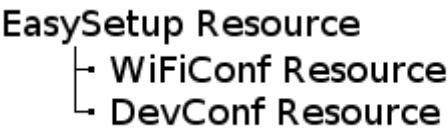
228 **6.1 개요**

229 Easy Setup이 가능한 Device는 다음과 같은 Resource Type을 지원해야 한다.

- 230 1. EasySetup Resource Type
- 231 2. WiFiConf Resource Type
- 232 3. DevConf Resource Type

233 EasySetup Resource Type은 Collection Resource로 최소한 WiFiConf와 DevConf의 인스턴스에
234 대한 Link를 포함해야 한다. 제조사는 그 밖의 Resource Type에 대한 link를 추가할 수 있다.

235 EasySetup Resource Type은 복수의 요청이 아닌 단일 요청으로 각각의 링크된 Resource로
236 효율적으로 데이터를 전달하도록 하는 batch Interface (oic.if.b)를 지원한다는 점에 유의하기 바란다.



237

238 **그림 2. Easy Setup Resource Type**

239 **6.2 EasySetup Resource**

240 **6.2.1 개략**

241 EasySetup Resource는 Enrollee의 현재 상태 및 Easy Setup 과정에서 발생한 마지막 에러 코드를
242 포함하는 유용한 정보를 저장한다.

243 **6.2.2 Resource**

244 Easy Setup Resource Type은 표 1. EasySetup Resource Type에 정의된 바와 같다.

245 **표 1. EasySetup Resource Type**

URI 예	Resource Type Title	Resource Type ID ("rt" 값)	Interface	설명	관련된 기능적 상호 작용
/example/EasySetupResourcesURI	EasySetup	oic.r.easysetup, oic.wk.col	oic.if.baseline, oic.if.ll, oic.if.b	Easy Setup용 최상위 레벨 Resource. easy setup 상태를 나타낸다. Resource property는 표 2에 나열된 바와 같다.	

246 표 2. “oic.r.easysetup” Resource Type 정의에서는 “oic.r.easysetup” Resource Type 에 대한 세부
 247 정보를 정의한다.

248 표 2. “oic.r.easysetup” Resource Type 정의

Property title	Property name	값 유형	값 규칙	단위	액세스 모드	필수	설명
Easy Setup Provisioning Status	ps	정수	enum		R	예	Device 의 Easy setup 프로비저닝 상태. 0: 설정 필요 1: Enroller 에 접속 중 2: Enroller 에 접속됨 3: Enroller 에 접속 실패 4~254: Reserved 255: EOF
Last Error Code	lec	정수	enum		R	예	Enroller 에의 접속 실패 시에 실패 원인을 가리킨다. 0: 에러 없음 1: 주어진 SSID 를 찾을 수 없다 2: Wi-Fi 패스워드가 올바르지 않다 3: IP 주소가 할당되어 있지 않다 4: 인터넷에 연결되어 있지 않다 5: 시간 초과 6: Wi-Fi Auth Type 이 Enrollee 에서 지원되지 않는다 7: Wi-Fi Encryption Type 이 Enrollee 에서 지원되지 않는다 8: Wi-Fi Auth Type 이 올바르지 않다 (Enroller 에 접속 중의 실패), 9: Wi-Fi Encryption Type 이 올바르지 않다 (Enroller 에 접속 중의 실패), 10~254: Reserved 255: 불명확한 에러

Property title	Property name	값 유형	값 규칙	단위	액세스 모드	필수	설명
Connect	cn	정수의 배열			RW	예	Enrollee 가 접속을 개시하도록 하는 접속 유형의 배열: 1: Wi-Fi 2: 추후 추가될 전송 방식 (예: BLE))
Links	links	배열			R	예	WiFiConf 및 DevConf Resource 인 링크의 배열.

249 Enrollee 는 기본 값으로 다음을 설정해야 한다 (예를 들어, Device 를 처음으로 개봉했을 때).

- 250 • "ps" = 0.
- 251 • "lec" = 0.
- 252 • "cn" = 빈 배열.

253 6.3 WiFiConf Resource Type

254 6.3.1 개요

255 WiFiConf Resource Type 은 Enrollee 가 기존의 Wi-Fi AP 에 접속하는데 필요한 정보를 저장한다.

256 6.3.2 Resource Type

257 WiFiConf Resource Type 은 표 3. WiFiConf Resource Type 에 정의된 바와 같다.

258 **표 3. WiFiConf Resource Type**

URI 예	Resource Type Title	Resource Type ID ("rt" 값)	Interface	설명	관련된 기능적 상호 작용
/example/WiFiConfResourceURI	WiFiConf	oic.r.wificonf	oic.if.baseline, oic.if.rw	Wi-Fi 관련 property 를 포함한다. Resource property 는 표 4 에 나열된 바와 같다.	

259 표 4. "oic.r.wificonf" Resource Type 정의에서는 "oic.r.wificonf" Resource Type 에 대한 세부 정보를
260 정의한다.

표 4. "oic.r.wificonf" Resource Type 정의

Property title	Property name	값 유형	값 규칙	단위	엑세스 모드	필수	설명
Supported Wi-Fi Mode Type	swmt	스트링의 배열	enum		R	예	Enrollee 에서 지원되는. Wi-Fi 모드. 복수의 모드일 수 있다. ("A", "B", "G", "N", "AC")
Supported Wi-Fi Frequency	swf	스트링의 배열	Refer to description for valid values.		R	예	Enrollee 에서 지원되는 Wi-Fi 주파수. 복수의 주파수일 수 있다. ("2.4G", "5G")
Target Network Name	tnn	스트링			RW	예	대상 네트워크명 (Wi-Fi AP, 즉 enroller 의 SSID).
Credential	cd	스트링			RW	아니오	Wi-Fi AP 의 크리덴셜 정보 (enroller 에 접속하는데 사용되는 비밀번호).
Wi-Fi Auth Type	wat	스트링	enum		RW	예	Wi-Fi auth type ("None", "WEP", "WPA_PSK", "WPA2_PSK")
Wi-Fi Encryption Type	wet	스트링	enum		RW	예	Wi-Fi encryption type ("None", "WEP_64", "WEP_128", "TKIP", "AES", "TKIP_AES")
Supported Wi-Fi Auth Type	swat	스트링의 배열	enum		R	예	지원되는 Wi-Fi Auth type. 복수의 타입일 수 있다. ("None", "WEP", "WPA_PSK", "WPA2_PSK")
Supported Wi-Fi Encryption Type	swet	스트링의 배열	enum		R	예	지원되는 Wi-Fi Encryption type. 복수의 타입일 수 있다. ("None", "WEP-64", "WEP_128", "TKIP", "AES", "TKIP_AES")

262 **6.4 DevConf Resource Type**263 **6.4.1 Introduction**

264 DevConf Resource Type 은 Wi-Fi Easy Setup 에 필요한 Device 구성 정보를 저장한다.

265

266

267

268

269

270

270

271

7 네트워크 및 연결성

Mediator 와 Enrollee 는 공유 연결 (예를 들어, Wi-Fi)을 통해 통신한다.

Wi-Fi 를 사용해서 Easy Setup 을 수행하는 경우 Enrollee 는 Soft AP 로 작용하는 기능을 가져야 한다. Soft AP 는 IEEE 802.11:2016 에 의해 정의된 액세스 포인트 요건을 지원해야 한다.

8 기능적 상호 작용

8.1 온보딩, 프로비저닝, 및 구성

Mediator 는 Enrollee 에 대한 소유권 이전을 수행할 수 있으며, ACL 프로비저닝을 수행할 수도 있다. 그러한 경우, Mediator 는 모든 기존의 DOXS 및 AMS 요건을 따라야 한다. 자세한 사항은 OCF Security 시방서를 참조하기 바란다.

8.2 Resource 탐색

Mediator 는 상호 지원되는 연결을 통해 Enrollee 에 접속한다.

Easy Setup 단계에서 Enrollee 와 Mediator 간의 접속에 Wi-Fi 를 사용하는 경우, Enrollee 는 Soft AP 로서 탐색 가능해야 한다. Soft AP 는 OCF Security 시방서에 문서화된 추가적인 이용 제약을 갖는다.

8.3 Easy Setup Resource 의 검색 및 업데이트

Enrollee 는 Mediator 가 표준 OCF Resource 탐색 방법을 사용해서 (즉, /oic/res 에서 RETRIEVE 를 통해) 탐색할 수 있도록 Easy Setup Resource 를 노출시켜야 한다 (OCF Core 시방서, 섹션 11.3 참조).

Easy Setup Resource 는 안전한 Endpoint (예를 들어, CoAPS)만 노출시켜야 한다; OCF Core 시방서, 섹션 10 참조.

Mediator 는 Easy Setup 의 어느 단계에서도 Enrollee 의 상태를 확인하기 위해 Easy Setup Resource 의 검색을 요청할 수 있다. 이것은 Enrollee 와 Mediator 가 공유 네트워크 상에 있을 때만 적용된다.

Mediator 는 Enrollee 가 Resource Property 를 업데이트하도록 요청할 수 있다. Mediator 로부터의 요청이 있으면, Enrollee 는 현재 Resource Property Value 를 업데이트해야 하며, 요구된 모든 동작을 수행해야 한다. 예를 들어, Mediator 에 의해 "EasySetup" Resource 의 "cn" Property 가 업데이트되어 Wi-Fi 에의 연결을 나타내면, Enrollee 는 Enroller 에의 접속을 시작해야 한다.

Easy Setup Resource 에 대한 자세한 사항은 섹션 6 을 참조하기 바란다.

8.4 에러 처리

EasySetup Resource (즉, oic.r.easysetup)의 "lec" Property 는 Enroller 에 (WiFiConf Resource 에서 Mediator 에 의해 제공된 정보를 사용하여) 연결 시의 Easy Setup 과정 중에 발생하는 에러를 나타내는데 사용된다.

- SSID 를 찾을 수 없어서 연결을 실패한 경우, Enrollee 는 "lec" Property 를 1 로 설정해야 한다.
- 잘못된 크리덴셜 (비밀번호) 정보로 인해 연결을 실패한 경우, Enrollee 는 "lec" Property 를 2 로 설정해야 한다.
- Enrollee 에서 Auth type 을 지원하지 않는 경우, Enrollee 는 "lec" Property 를 6 으로 설정해야 한다.
- Enrollee 에서 Encryption type 을 지원하지 않는 경우, Enrollee 는 "lec" Property 를 7 로 설정해야 한다.
- 잘못된 Auth type 정보로 인해 연결을 실패한 경우 (Enrollee 에서 지원하더라도), Enrollee 는 "lec" Property 를 8 로 설정해야 한다.
- 잘못된 Encryption type 정보로 인해 연결을 실패한 경우 (Enrollee 에서 지원하더라도), Enrollee 는 "lec" Property 를 9 로 설정해야 한다.

Enrollee 와 Mediator 간의 연결에 Wi-Fi 를 사용하는 경우에 Enrollee 가 Enroller 에의 연결을 실패했을 때는, Enrollee 는 다시 자신을 Soft AP 로 탐색 가능하도록 해야 한다 (이전에 Soft AP 를 파기한 경우).

8.5 Easy Setup Flow 예

그림 3 은 정보 제공을 목적으로 한 Easy Setup 흐름의 예를 보여준다.

Easy Setup Flow (Informative)

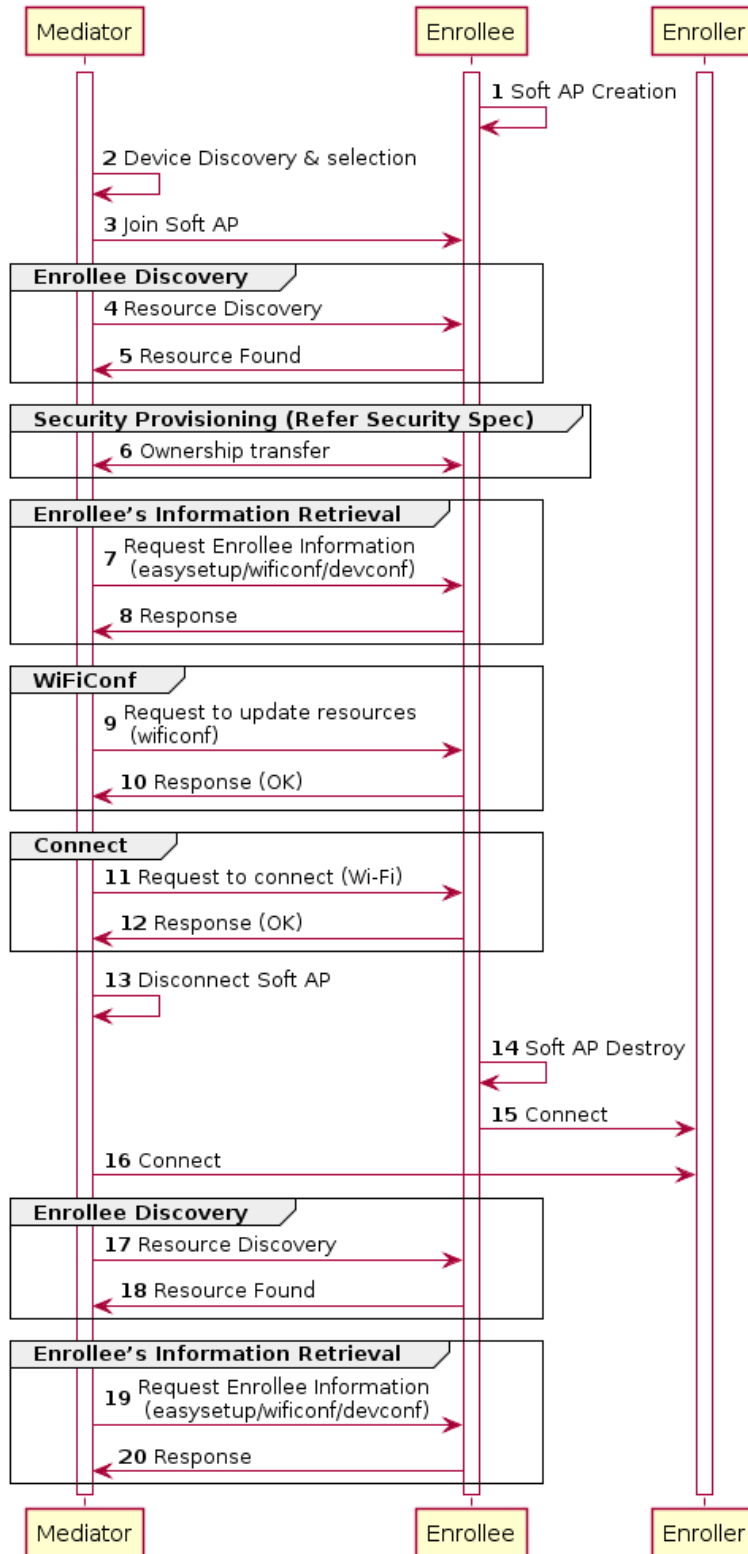


그림 3. Easy Setup Flow (참고)

323 위의 흐름 예에서는 Easy Setup 중에 보안 프로비저닝(6 단계)을 거친다. 원하는 경우에는 Enrollee
324 Discovery (4 및 5 단계) 전에 보안 프로비저닝을 수행할 수도 있다. 다른 시나리오에 관한 더 자세한
325 사항은 OCF Security 시방서를 참조하기 바란다.

326 **9 보안**

327 Wi-Fi Easy Setup 보안 요건은 OCF Security 시방서에서 찾을 수 있다.

Annex A (규정)

Resource Type 정의

A.1 Resource Type 정의 목록

표 7 은 이 시방서에서 정의된 resource 목록을 보여준다.

표 7. 알파벳순 resource 목록

명칭 (단순 정보)	Resource Type (rt)	섹션
Easy Setup	"oic.r.easysetup"	A.2
Wi-Fi Configuration	"oic.r.wificonf"	A.3
Device Configuration	"oic.r.devconf"	A.4

A.2 Easy Setup Collection Baseline Interface

A.2.1 개요

Easy Setup resource 는 개봉하는 device 의 현재 상태 및 easy setup 과정에서 발생한 에러 코드를 포함하는 유용한 정보를 저장한다. Easy Setup resource 는 WiFiConf 및 DevConf resource 에의 링크를 포함하고 그 밖의 resource 에의 링크를 추가로 포함할 수 있는 일종의 collection resource 임에 유의하기 바란다.

A.2.2 URI 예

/example/EasySetupBaselineInterfaceResURI

A.2.3 Resource Type

resource type (rt)는 oic.r.easysetup 으로 정의된다.

A.2.4 RAML 정의

##RAML 0.8

title: Easy Setup Resource

version: v0.0.3-20170611

traits:

- interface-II :

queryParameters:

```

351         if:
352             enum: ["oic.if.ll"]
353     - interface-baseline :
354         queryParameters:
355             if:
356                 enum: ["oic.if.baseline"]
357     - interface-all :
358         queryParameters:
359             if:
360                 enum: ["oic.if.baseline", "oic.if.ll", "oic.if.b"]
361     - interface-batch :
362         queryParameters:
363             if:
364                 enum: ["oic.if.b"]
365
366     /example/EasySetupBaselineInterfaceResURI:
367         description: |
368             Easy Setup resource stores useful information including current status of
369             unboxing device and last error code which are produced in a process of
370             easy setup.
371             Note that, Easy Setup resource is a type of collection resource, which
372             contains links to WiFiConf, DevConf resources and may additionally contain
373             links to other resources.
374
375         is : ["interface-baseline"]
376         get:
377             description: |
378                 Retrieve useful information during easy setup process :
379                 1
380                 A current status in easy setup process.
381                 2
382                 A last error code describing reason for failure occurred at the last
383                 time.
384
385         responses :
386             200:
387                 body:
388                     application/json:
389                         schema: /

```

```

390     {
391         "$schema": "http://json-schema.org/draft-04/schema#",
392         "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
393         "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.r.easyssetup-schema.json#",
394         "definitions": {
395             "oic.r.easyssetup": {
396                 "type": "object",
397                 "allOf": [
398                     {
399                         "$ref": "oic.collection-schema.json#/definitions/oic.collection"
400                     },
401                     {
402                         "properties": {
403                             "rt": {
404                                 "type": "array",
405                                 "minItems": 2,
406                                 "maxItems": 2,
407                                 "uniqueItems": true,
408                                 "items": {
409                                     "enum": ["oic.r.easyssetup", "oic.wk.col"]
410                                 }
411                             },
412                             "ps": {
413                                 "type": "integer",
414                                 "enum": [0, 1, 2, 3],
415                                 "description": "Indicates the easy setup status of the device. (0: Need to Setup, 1: Connecting to Enroller, 2:
416 Connected to Enroller, 3: Failed to Connect to Enroller, 4~254: Reserved, 255: EOF)",
417                                 "readOnly": true
418                             },
419                             "lec": {
420                                 "type": "integer",
421                                 "enum": [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 255],
422                                 "description": "Indicates a failure reason (0: NO error, 1: A given SSID is not found, 2: Wi-Fi's password is
423 wrong, 3: IP address is not allocated, 4: No internet connection, 5: Timeout, 6: Wi-Fi Auth Type is not supported by the Enrollee, 7:
424 Wi-Fi Encryption Type is not supported by the Enrollee, 8: Wi-Fi Auth Type is wrong (failure while connecting to the Enroller), 9:
425 Wi-Fi Encryption Type is wrong (failure while connecting to the Enroller), 10~254: Reserved, 255: Unknown error)",
426                                 "readOnly": true
427                             },
428                             "cn": {
429                                 "type": "array",
430                                 "description": "Indicates an array of connection types that trigger an attempt to connect to the Enroller to
431 start.",
432                                 "items": {
433                                     "type": "integer",
434                                     "description": "Connection type to attempt. (1 : Wi-Fi, 2 : other entities / transports to be added in future
435 (e.g. Connect to cloud / BLE))"

```

```

436         }
437     }
438 },
439     "required": ["ps", "lec", "cn"]
440 }
441 ]
442 }
443 },
444 "type": "object",
445 "allOf": [
446     { "$ref": "oic.core-schema.json#/definitions/oic.core"},
447     { "$ref": "#/definitions/oic.r.easyssetup" }
448 ]
449 }

```

450

451 example: /

```

452 {
453     "rt" : ["oic.r.easyssetup", "oic.wk.col"],
454     "if" : ["oic.if.ll", "oic.if.baseline", "oic.if.b"],
455     "ps" : 0,
456     "lec": 0,
457     "cn": [1],
458     "links": [
459         {
460             "href": "/EasySetupResURI",
461             "rt": ["oic.r.easyssetup", "oic.wk.col"],
462             "if": ["oic.if.b"],
463             "p":{"bm":3},
464             "eps": [
465                 {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2}
466             ],
467             "rel":["self", "item"]
468         },
469         {
470             "href": "/WiFiConfResURI",
471             "rt": ["oic.r.wificonf"],
472             "if": ["oic.if.baseline"],
473             "p":{"bm":3},
474             "eps": [
475                 {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2}
476             ]
477         },
478         {
479             "href": "/DevConfResURI",
480             "rt": ["oic.r.devconf"],
481             "if": ["oic.if.baseline"],

```



```

482         "p":{"bm":3},
483         "eps": [
484             {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2}
485         ]
486     }
487 ]
488 }

```

489 A.2.5 Property 정의

Property 명칭	값 유형	필수	액세스 모드	설명
rt	배열: schema 참조		Read Write	
ps	정수	예	Read Write	device 의 easy setup 상태를 나타낸다. (0: Setup 필요, 1: Enroller 에 연결 중, 2: Enroller 에 연결됨, 3: Enroller 에 연결 실패, 4~254: Reserved, 255: EOF)
lec	정수	예	Read Write	연결 실패의 원인을 나타낸다 (0: 에러 없음, 1: 주어진 SSID 를 찾을 수 없다, 2: Wi-Fi 의 비밀번호 불일치, 3: IP 주소가 할당되어 있지 않다, 4: 인터넷에 연결되어 있지 않다, 5: 시간 초과, 6: Enrollee 에서 Wi-Fi Auth Type 을 지원하지 않는다, 7: Enrollee 에서 Wi-Fi Encryption Type 을 지원하지 않는다, 8: Wi-Fi Auth Type 이 맞지 않는다 (Enroller 에 연결 중의 실패), 9: Wi-Fi Encryption Type 이 맞지 않는다 (Enroller 에 연결 중의 실패), 10~254: Reserved, 255: 불명확한 에러)
cn	배열: schema 참조	예	Read Write	Enroller 에의 연결 시도를 시작하도록 하는 연결 유형의 배열을 나타낸다.

490 A.2.6 CRUDN 동작

Resource	Create	Read	Update	Delete	Notify
/example/EasySetupBaselineInterfaceResURI		get			

491 A.3 Wi-Fi Configuration Resource Baseline Interface

492 A.3.1 개요

493 WiFiConf resource 는 개봉하는 device 가 기존의 Wi-Fi AP 에 연결하기 위한 필수 정보를 저장한다.

```

494 A.3.2 URI 예
495 /example/WiFiConfBaselineInterfaceResURI
496 A.3.3 Resource Type
497 resource type (r 는 oic.r.wificonf 로 정의된다.
498 A.3.4 RAML 정의
499 #%RAML 0.8
500 title: Wi-Fi Configuration Resource
501 version: v0.0.3-20170611
502 traits:
503   - interface-rw :
504     queryParameters:
505       if:
506         enum: ["oic.if.rw"]
507   - interface-baseline :
508     queryParameters:
509       if:
510         enum: ["oic.if.baseline"]
511   - interface-all :
512     queryParameters:
513       if:
514         enum: ["oic.if.baseline", "oic.if.rw"]
515
516 /example/WiFiConfBaselineInterfaceResURI:
517   description: |
518     WiFiConf resource stores essential information to help an unboxing device
519     to connect to an existing Wi-Fi AP.
520
521   is : ['interface-baseline']
522   get:
523     description: |
524       Retrieve properties of WiFiConf resource.
525       The information includes :
526       1
527       Wi-Fi SSID and password
528       2
529       Wi-Fi Security type (i.e
530       auth type and encryption type)
531       3

```

```

532     Wi-Fi hardware capability (i.e
533     supported frequencies, modes,
534     auth types and encryption types)
535
536     responses :
537     200:
538     body:
539     application/json:
540     schema: /
541     {
542     "$schema": "http://json-schema.org/draft-04/schema#",
543     "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
544     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.r.wificonf-schema.json#",
545     "definitions": {
546     "oic.r.wificonf": {
547     "type": "object",
548     "properties": {
549     "swmt": {
550     "type": "array",
551     "description": "Indicates supported Wi-Fi mode types. It can be multiple",
552     "readOnly": true,
553     "items":
554     {
555     "type": "string",
556     "enum": ["A", "B", "G", "N", "AC"],
557     "description": "Supported Wi-Fi Mode Type."
558     }
559     },
560     "swf": {
561     "type": "array",
562     "description": "Indicates Supported Wi-Fi frequencies by the Enrollee. Can be multiple. Valid values are ('2.4G',
563     '5G')",
564     "readOnly": true,
565     "items":
566     {
567     "type": "string",
568     "pattern": "^((2|4|5)G)$"
569     }
570     },
571     "tnn": {
572     "type": "string",
573     "description": "Indicates Target Network Name (SSID of Wi-Fi AP)",
574     "pattern": "^[a-zA-Z0-9_]{1,32}$"
575     },
576     "cd": {

```

```

577         "type": "string",
578         "description": "Indicates credential information of Wi-Fi AP",
579         "pattern": "^.*$"
580     },
581     "wat": {
582         "type": "string",
583         "enum": ["None", "WEP", "WPA_PSK", "WPA2_PSK"],
584         "description": "Indicates Wi-Fi Auth Type"
585     },
586     "wet": {
587         "type": "string",
588         "enum": ["None", "WEP_64", "WEP_128", "TKIP", "AES", "TKIP_AES"],
589         "description": "Indicates Wi-Fi Encryption Type"
590     },
591     "swat": {
592         "type": "array",
593         "description": "Indicates supported Wi-Fi Auth types. It can be multiple",
594         "readOnly": true,
595         "items":
596         {
597             "type": "string",
598             "enum": ["None", "WEP", "WPA_PSK", "WPA2_PSK"],
599             "description": "Indicates Wi-Fi Auth Type"
600         }
601     },
602     "swet": {
603         "type": "array",
604         "description": "Indicates supported Wi-Fi Encryption types. It can be multiple",
605         "readOnly": true,
606         "items":
607         {
608             "type": "string",
609             "enum": ["None", "WEP_64", "WEP_128", "TKIP", "AES", "TKIP_AES"],
610             "description": "Indicates Wi-Fi Encryption Type"
611         }
612     },
613 },
614 "required":["swmt", "swf", "swat", "swet", "tnn", "wat", "wet"]
615 }
616 },
617 "type": "object",
618 "allOf": [
619     { "$ref": "oic.core-schema.json#/definitions/oic.core"},
620     { "$ref": "#/definitions/oic.r.wificonf" }
621 ]

```

```

622     }
623
624     example: /
625     {
626         "rt": ["oic.r.wificonf"],
627         "swmt": ["A", "B", "G"],
628         "swf": ["2.4G", "5G"],
629         "tnn": "Home_AP_SSID",
630         "cd": "Home_AP_PWD",
631         "wat": "WPA2_PSK",
632         "wet": "TKIP",
633         "swat": ["WPA_PSK", "WPA2_PSK"],
634         "swet": ["TKIP", "AES", "TKIP_AES"]
635     }
636
637     post:
638         description: |
639             Deliver Wi-Fi AP's information for an unboxing device to connect to it.
640
641         body:
642             application/json:
643                 schema: /
644                 {
645                     "$schema": "http://json-schema.org/draft-v4/schema#",
646                     "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
647                     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.r.wificonf-update-schema.json#",
648                     "definitions": {
649                         "oic.r.wificonf": {
650                             "type": "object",
651                             "properties": {
652                                 "tnn": {
653                                     "type": "string",
654                                     "description": "Indicates Target Network Name (SSID of Wi-Fi AP)",
655                                     "pattern": "^.*$"
656                                 },
657                                 "cd": {
658                                     "type": "string",
659                                     "description": "Indicates credential information of Wi-Fi AP",
660                                     "pattern": "^.*$"
661                                 },
662                                 "wat": {
663                                     "enum": ["None", "WEP", "WPA_PSK", "WPA2_PSK"],
664                                     "description": "Indicates Wi-Fi Auth Type"
665                                 }

```

```

666         "wet": {
667             "enum": ["None", "WEP_64", "WEP_128", "TKIP", "AES", "TKIP_AES"],
668             "description": "Indicates Wi-Fi Encryption Type"
669         }
670     },
671     "required":["tnn", "wat", "wet"]
672 }
673 },
674 "type": "object",
675 "allOf": [
676     { "$ref": "oic.core-schema.json#/definitions/oic.core"},
677     { "$ref": "#/definitions/oic.r.wificonf" }
678 ]
679 }
680
681 example: /
682 {
683     "tnn": "Home_AP_SSID",
684     "cd": "Home_AP_PWD",
685     "wat": "WPA2_PSK",
686     "wet": "AES"
687 }
688
689 responses :
690 200:
691 body:
692 application/json:
693 schema: /
694 {
695     "$schema": "http://json-schema.org/draft-v4/schema#",
696     "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
697     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.r.wificonf-update-schema.json#",
698     "definitions": {
699         "oic.r.wificonf": {
700             "type": "object",
701             "properties": {
702                 "tnn": {
703                     "type": "string",
704                     "description": "Indicates Target Network Name (SSID of Wi-Fi AP)",
705                     "pattern": "^.*$"
706                 },
707                 "cd": {
708                     "type": "string",
709                     "description": "Indicates credential information of Wi-Fi AP",

```

```

710         "pattern": "^.*$"
711     },
712     "wat": {
713         "enum": ["None", "WEP", "WPA_PSK", "WPA2_PSK"],
714         "description": "Indicates Wi-Fi Auth Type"
715     },
716     "wet": {
717         "enum": ["None", "WEP_64", "WEP_128", "TKIP", "AES", "TKIP_AES"],
718         "description": "Indicates Wi-Fi Encryption Type"
719     }
720 },
721 "required":["tnn", "wat", "wet"]
722 }
723 },
724 "type": "object",
725 "allOf": [
726     { "$ref": "oic.core-schema.json#/definitions/oic.core"},
727     { "$ref": "#/definitions/oic.r.wificonf" }
728 ]
729 }

```

731 example: /

```

732 {
733     "tnn": "Home_AP_SSID",
734     "cd": "Home_AP_PWD",
735     "wat": "WPA2_PSK",
736     "wet": "AES"
737 }

```

738

739 **A.3.5 Property 정의**

Property 명칭	값 유형	필수	액세스 모드	설명
tnn	스트링	예	Read Write	대상 네트워크 명칭을 나타낸다 (Wi-Fi AP 의 SSID)
swmt	배열: schema 참조	예	Read Only	지원되는 Wi-Fi 모드 유형을 나타낸다. 복수의 유형일 수 있다.
swat	배열: schema 참조	예	Read Only	지원되는 Wi-Fi Auth 유형을 나타낸다. 복수의 유형일 수 있다.
cd	스트링		Read Write	Wi-Fi AP 의 크리덴셜 정보를 나타낸다.
swf	배열: schema 참조	예	Read Only	Enrollee 에서 지원하는 Wi-Fi 주파수를 나타낸다. 복수의 주파수일 수 있다. 유효한 값은 ('2.4G', '5G')이다.
wet	스트링	예	Read Write	Wi-Fi Encryption Type 을 나타낸다.
wat	스트링	예	Read Write	Wi-Fi Auth Type 을 나타낸다.
swet	배열: schema 참조	예	Read Only	지원되는 Wi-Fi Encryption 유형을 나타낸다. 복수의 유형일 수 있다.

740 **A.3.6 CRUDN 동작**

Resource	Create	Read	Update	Delete	Notify
/example/WiFiConfBaselineInterfaceResURI		get	post		

741 **A.4 Device Configuration**

742 **A.4.1 개요**

743 Device configuration resource 는 device 명칭과 같은 기본 device 설정을 저장한다. 제조사 특정
744 정보를 resource 에 추가할 수 있다.


```

745 A.4.2 URI 예
746 /example/DevConfResURI
747 A.4.3 Resource Type
748 resource type (rt)는 oic.r.devconf 로 정의된다.
749 A.4.4 RAML 정의
750 #%RAML 0.8
751 title: Device Configuration Resource
752 version: v0.0.2-20170604
753 traits:
754 - interface :
755     queryParameters:
756         if:
757             enum: ["oic.if.baseline", "oic.if.r"]
758
759 /example/DevConfResURI:
760     description: |
761         Device configuration resource stores a preference of device settings like
762         device name
763         Vender-specific information can be added to the resource.
764
765     is : ['interface']
766     get:
767         description: |
768             Retrieve various settings regarding to device-specific settings
769             1
770             Device name (human-friendly name to be detected by mediator during
771             easy setup)
772
773     responses :
774         200:
775             body:
776                 application/json:
777                     schema: /
778                     {
779                         "$schema": "http://json-schema.org/draft-04/schema#",
780                         "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
781                         "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.r.devconf-schema.json#",
782                         "definitions": {

```

```

783     "oic.r.devconf": {
784         "type": "object",
785         "oneOf": [
786             {
787                 "properties": {
788                     "dn": {
789                         "type": "string",
790                         "description": "Indicates a pre-configured device name in language indicated by 'dl' in /oic/con; presented by
791 enrollee device to mediator device during easy-setup process",
792                         "pattern": "^.*$",
793                         "readOnly": true
794                     }
795                 },
796                 "required": ["dn"]
797             },
798             {
799                 "properties": {
800                     "dn": {
801                         "type": "array",
802                         "items": {
803                             "type": "object",
804                             "properties": {
805                                 "language": {
806                                     "$ref": "oic.types-schema.json#/definitions/language-tag",
807                                     "readOnly": true,
808                                     "description": "An RFC 5646 language tag."
809                                 },
810                                 "value": {
811                                     "type": "string",
812                                     "description": "Pre-configured device name in the indicated language.",
813                                     "pattern": "^.*$",
814                                     "readOnly": true
815                                 }
816                             }
817                         },
818                         "minItems": 1,
819                         "readOnly": true,
820                         "description": "Localized device name."
821                     }
822                 },
823                 "required": ["dn"]
824             }
825         ]
826     },
827 },
828 "type": "object",

```

```

829     "allOf": [
830       { "$ref": "oic.core-schema.json#/definitions/oic.core"},
831       { "$ref": "#/definitions/oic.r.devconf" }
832     ]
833   }
834

```

```

835   example: /
836   {
837     "rt": ["oic.r.devconf"],
838     "dn" : "My Refrigerator"
839   }
840

```

841 A.4.5 Property Definition

Property 명칭	값 유형	필수	액세스 모드	설명
dn	배열: schema 참조	예	Read Only	로컬 device 명칭.

842 A.4.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/example/DevConfResURI		get			

843

Annex B (참고)

Swagger2.0 정의

B.1 Device 구성

B.1.1 개요

Device configuration resource 는 device 명칭과 같은 기본 device 설정을 저장한다. 제조사 특정 정보를 resource 에 추가할 수 있다.

device 특정 설정에 관한 다양한 설정을 검색한다.

1. Device 명칭 (easy setup 중에 mediator 에 의해 검출되는 인식하기 쉬운 명칭)

B.1.2 잘 알려진 URI

/example/DevConfResURI

B.1.3 Resource Type

resource type (rt)는 ['oic.r.devconf']로 정의된다.

B.1.4 Swagger2.0 정의

```
{
  "swagger": "2.0",
  "info": {
    "title": "Device Configuration",
    "version": "v0.0.2-20170604",
    "license": {
      "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
      "x-description": "Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:\n      1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.\n      2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open Connectivity Foundation, INC. \n\n      AS IS\n\n      AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n\n      IN NO EVENT SHALL THE Open Connectivity Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n\n      HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.\n\n"
```

```

879 },
880 "schemes": ["http"],
881 "consumes": ["application/json"],
882 "produces": ["application/json"],
883 "paths": {
884     "/example/DevConfResURI" : {
885         "get": {
886             "description": "Device configuration resource stores a preference of device settings like device name. Vender-specific
887 information can be added to the resource. Retrieve various settings regarding to device-specific settings. 1. Device name
888 (human-friendly name to be detected by mediator during easy setup)",
889             "parameters": [
890                 {"$ref": "#/parameters/interface"}
891             ],
892             "responses": {
893                 "200": {
894                     "description": "",
895                     "x-example":
896                     {
897                         "rt": ["oic.r.devconf"],
898                         "dn" : "My Refrigerator"
899                     }
900                 },
901                 "schema": { "$ref": "#/definitions/DevConf" }
902             }
903         }
904     }
905 },
906 },
907 "parameters": {
908     "interface" : {
909         "in" : "query",
910         "name" : "if",
911         "type" : "string",
912         "enum" : ["oic.if.baseline", "oic.if.r"]
913     }
914 },
915 "definitions": {
916     "DevConf" :
917     {
918         "oneOf": [
919             {
920                 "properties": {
921                     "dn": {
922                         "description": "Indicates a pre-configured device name in language indicated by 'dl' in /oic/con; presented by enrollee
923 device to mediator device during easy-setup process",
924                         "pattern": "^[.*$]",

```

```

925         "readOnly": true,
926         "type": "string"
927     }
928 },
929     "required": [
930         "dn"
931     ]
932 },
933 {
934     "properties": {
935         "dn": {
936             "description": "Localized device name.",
937             "items": {
938                 "properties": {
939                     "language": {
940                         "description": "An RFC 5646 language tag.",
941                         "pattern": "^[A-Za-z]{1,8}(-[A-Za-z0-9]{1,8})*$",
942                         "readOnly": true,
943                         "type": "string"
944                     },
945                     "value": {
946                         "description": "Pre-configured device name in the indicated language.",
947                         "pattern": "^.*$",
948                         "readOnly": true,
949                         "type": "string"
950                     }
951                 },
952                 "type": "object"
953             },
954             "minItems": 1,
955             "readOnly": true,
956             "type": "array"
957         }
958     },
959     "required": [
960         "dn"
961     ]
962 }
963 ],
964     "type": "object"
965 }
966 }
967 }
968 }
969

```

970 B.1.5 Property Definition

Property 명칭	값 유형	필수	액세스 모드	설명
dn	배열: schema 참조	예	Read Only	로컬 device 명칭.

971 B.1.6 CRUDN behaviour

Resource	Create	Read	Update	Delete	Notify
/example/DevConfResURI		get			

972 B.2 Easy Setup Collection

973 B.2.1 개요

974 Easy Setup resource 는 개봉하는 device 의 현재 상태 및 easy setup 과정에서 발생한 에러 코드를
975 포함하는 유용한 정보를 저장한다.

976 Easy Setup resource 는 WiFiConf 및 DevConf resource 에의 링크를 포함하고 그 밖의
977 resource 에의 링크를 추가로 포함할 수 있는 일종의 collection resource 임에 유의하기 바란다.

978 easy setup 과정 중에 유용한 정보를 검색한다.

- 979 1. easy setup 과정의 현재 상태.
- 980 2. 마지막으로 발생한 실패의 원인을 기술하는 마지막 에러 코드.

981 B.2.2 잘 알려진 URI

982 /example/EasySetupBaselineInterfaceResURI

983 B.2.3 Resource Type

984 resource type (rt)는 ['oic.r.easyssetup', 'oic.wk.col']로 정의된다.

985 B.2.4 Swagger2.0 정의

```

986 {
987   "swagger": "2.0",
988   "info": {
989     "title": "Easy Setup Collection Batch Interface",
990     "version": "v0.0.3-20170611",
991     "license": {
992       "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
993       "x-description": "Redistribution and use in source and binary forms, with or without modification, are permitted provided that
994 the following conditions are met:\n      1. Redistributions of source code must retain the above copyright notice, this list of
995 conditions and the following disclaimer.\n      2. Redistributions in binary form must reproduce the above copyright notice, this
996 list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.\n\n
997 THIS SOFTWARE IS PROVIDED BY THE Open Connectivity Foundation, INC. \nAS IS\n" AND ANY EXPRESS OR IMPLIED
998 WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A

```

```

999 PARTICULAR PURPOSE OR WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.
1000 IN NO EVENT SHALL THE Open
1001 Connectivity Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
1002 CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
1003 USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
1004 HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
1005 WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
1006 THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1007
1008 },
1009 "schemes": ["http"],
1010 "consumes": ["application/json"],
1011 "produces": ["application/json"],
1012 "paths": {
1013     "/example/EasySetupBatchInterfaceResURI" : {
1014         "get": {
1015             "description": "Easy Setup resource stores useful information including current status of
1016             code which are produced in a process of
1017             Easy Setup resource is a type of collection resource,
1018             which
1019             contains links to WiFiConf, DevConf resources and may additionally contain
1020             links to other resources.
1021             Retrieve useful
1022             information during easy setup process :
1023             1. A current status in easy setup process.
1024             2. A last error code describing reason for
1025             failure occurred at the last
1026             time.
1027
1028             "parameters": [
1029                 {
1030                     "$ref": "#/parameters/interface-batch"
1031                 }
1032             ],
1033             "responses": {
1034                 "200": {
1035                     "description": "",
1036                     "x-example":
1037                     [
1038                         {
1039                             "href": "/EasySetupResURI",
1040                             "rep": {
1041                                 "ps" : 0,
1042                                 "lec": 0,
1043                                 "cn": [1]
1044                             }
1045                         },
1046                         {
1047                             "href": "/WiFiConfResURI",
1048                             "rep": {
1049                                 "swmt" : ["A", "B", "G"],
1050                                 "swf": ["2.4G", "5G"],
1051                                 "tnn": "Home_AP_SSID",
1052                                 "cd": "Home_AP_PWD",
1053                                 "wat": "WPA2_PSK",
1054                                 "wet": "AES"
1055                             }
1056                         }
1057                     ]
1058                 }
1059             }
1060         }
1061     }
1062 }

```



```

1045         {
1046             "href": "/DevConfResURI",
1047             "rep":{
1048                 "dn" : "My Refrigerator"
1049             }
1050         }
1051     ]
1052     ,
1053     "schema": { "$ref": "#/definitions/sbatch" }
1054 }
1055 },
1056 ],
1057 "post": {
1058     "description": "Able to deliver Wi-Fi, Device configuration and other configuration information in a batch by utilizing
1059 'batch' interface. If you want to deliver Wi-Fi and Device configuration information in a batch, you can write all properties you
1060 want to send with a 'batch' interface. The below example is the case to send Easy Setup and Wi-Fi configuration (i.e.
1061 connection type, target network, auth type information) in a batch.",
1062     "parameters": [
1063         { "$ref": "#/parameters/interface-batch",
1064         {
1065             "name": "body",
1066             "in": "body",
1067             "required": true,
1068             "schema": { "$ref": "#/definitions/sbatch-update" },
1069             "x-example":
1070             [
1071                 {
1072                     "href": "/EasySetupResURI",
1073                     "rep":{
1074                         "cn": [1]
1075                     }
1076                 },
1077                 {
1078                     "href": "/WiFiConfResURI",
1079                     "rep":{
1080                         "tnn": "Home_AP_SSID",
1081                         "cd": "Home_AP_PWD",
1082                         "wat": "WPA2_PSK",
1083                         "wet": "AES"
1084                     }
1085                 }
1086             ]
1087             }
1088         ],
1089         "responses": {
1090             "200": {

```

```

1091         "description" : "",
1092         "x-example":
1093         [
1094             {
1095                 "href": "/EasySetupResURI",
1096                 "rep" : {
1097                     "ps" : 0,
1098                     "lec": 0,
1099                     "cn": [1]
1100                 }
1101             },
1102             {
1103                 "href": "/WiFiConfResURI",
1104                 "rep" : {
1105                     "swmt" : ["A", "B", "G"],
1106                     "swf": ["2.4G", "5G"],
1107                     "tnn": "Home_AP_SSID",
1108                     "cd": "Home_AP_PWD",
1109                     "wat": "WPA2_PSK",
1110                     "wet": "AES"
1111                 }
1112             },
1113             {
1114                 "href": "/DevConfResURI",
1115                 "rep" : {
1116                     "dn" : "My Refrigerator"
1117                 }
1118             }
1119         ]
1120     ,
1121     "schema": { "$ref": "#/definitions/sbatch" }
1122 }
1123 }
1124 }
1125 },
1126 "/example/EasySetupLLInterfaceResURI" : {
1127     "get": {
1128         "description": "Easy Setup resource stores useful information including current status of\nunboxing device and last error
1129 code which are produced in a process of\neasy setup.\nNote that, Easy Setup resource is a type of collection resource,
1130 which\ncontains links to WiFiConf, DevConf resources and may additionally contain\nlinks to other resources.\nRetrieve useful
1131 information during easy setup process :\n1. A current status in easy setup process.\n2. A last error code describing reason for
1132 failure occurred at the last\n   time.\n",
1133         "parameters": [
1134             {"$ref": "#/parameters/interface-ll"}
1135         ],
1136         "responses": {

```

```

1137     "200": {
1138         "description": "",
1139         "x-example":
1140         [
1141             {
1142                 "href": "/EasySetupResURI",
1143                 "rt": ["oic.r.easysetup", "oic.wk.col"],
1144                 "if": ["oic.if.b"],
1145                 "p":{"bm":3},
1146                 "eps": [
1147                     {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2}
1148                 ],
1149                 "rel":["self", "item"]
1150             },
1151             {
1152                 "href": "/WiFiConfResURI",
1153                 "rt": ["oic.r.wificonf"],
1154                 "if": ["oic.if.baseline"],
1155                 "p":{"bm":3},
1156                 "eps": [
1157                     {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2}
1158                 ]
1159             },
1160             {
1161                 "href": "/DevConfResURI",
1162                 "rt": ["oic.r.devconf"],
1163                 "if": ["oic.if.baseline"],
1164                 "p":{"bm":3},
1165                 "eps": [
1166                     {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2}
1167                 ]
1168             }
1169         ]
1170     ,
1171     "schema": { "$ref": "#/definitions/slinks" }
1172 }
1173 }
1174 }
1175 },

```

```

1176 "/example/EasySetupBaselineInterfaceResURI" : {
1177     "get": {

```

```

1178         "description": "Easy Setup resource stores useful information including current status of unboxing device and last error
1179 code which are produced in a process of easy setup. Note that, Easy Setup resource is a type of collection resource,
1180 which contains links to WiFiConf, DevConf resources and may additionally contain links to other resources. Retrieve useful
1181 information during easy setup process :
1182     1. A current status in easy setup process.
1183     2. A last error code describing reason for failure occurred at the last time."

```

```

1183     "parameters": [
1184         {"$ref": "#/parameters/interface-baseline"}
1185     ],
1186     "responses": {
1187         "200": {
1188             "description": "",
1189             "x-example":
1190             {
1191                 "rt": ["oic.r.easyssetup", "oic.wk.col"],
1192                 "if": ["oic.if.ll", "oic.if.baseline", "oic.if.b"],
1193                 "ps": 0,
1194                 "lec": 0,
1195                 "cn": [1],
1196                 "links": [
1197                     {
1198                         "href": "/EasySetupResURI",
1199                         "rt": ["oic.r.easyssetup", "oic.wk.col"],
1200                         "if": ["oic.if.b"],
1201                         "p":{"bm":3},
1202                         "eps": [
1203                             {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2}
1204                         ],
1205                         "rel":["self", "item"]
1206                     },
1207                     {
1208                         "href": "/WiFiConfResURI",
1209                         "rt": ["oic.r.wificonf"],
1210                         "if": ["oic.if.baseline"],
1211                         "p":{"bm":3},
1212                         "eps": [
1213                             {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2}
1214                         ]
1215                     },
1216                     {
1217                         "href": "/DevConfResURI",
1218                         "rt": ["oic.r.devconf"],
1219                         "if": ["oic.if.baseline"],
1220                         "p":{"bm":3},
1221                         "eps": [
1222                             {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2}
1223                         ]
1224                     }
1225                 ]
1226             }
1227         },
1228         "schema": { "$ref": "#/definitions/EasySetup" }

```

```

1229     }
1230 }
1231 }
1232 }
1233 },
1234 "parameters": {
1235     "interface-ll" : {
1236         "in" : "query",
1237         "name" : "if",
1238         "type" : "string",
1239         "enum" : ["oic.if.ll"]
1240     },
1241     "interface-baseline" : {
1242         "in" : "query",
1243         "name" : "if",
1244         "type" : "string",
1245         "enum" : ["oic.if.baseline"]
1246     },
1247     "interface-all" : {
1248         "in" : "query",
1249         "name" : "if",
1250         "type" : "string",
1251         "enum" : ["oic.if.baseline", "oic.if.ll", "oic.if.b"]
1252     },
1253     "interface-batch" : {
1254         "in" : "query",
1255         "name" : "if",
1256         "type" : "string",
1257         "enum" : ["oic.if.b"]
1258     }
1259 },
1260 "definitions": {
1261     "sbatch" :
1262     {
1263         "items": {
1264             "additionalProperties": true,
1265             "properties": {
1266                 "href": {
1267                     "description": "URI of the target resource relative assuming the collection URI as anchor",
1268                     "format": "uri",
1269                     "maxLength": 256,
1270                     "type": "string"
1271                 },
1272                 "rep": {
1273                     "oneOf": [
1274

```

```

1275         "description": "The response payload from a single resource",
1276         "type": "object"
1277     },
1278     {
1279         "description": " The response payload from a collection (batch) resource",
1280         "type": "array"
1281     }
1282 ]
1283 }
1284 },
1285 "required": [
1286     "href",
1287     "rep"
1288 ],
1289 "type": "object"
1290 },
1291 "minItems": 1,
1292 "type": "array"
1293 }
1294
1295 ,
1296 "sbatch-update" :
1297 {
1298     "description": "array of resource representations to apply to the batch collection, using href to indicate which resource(s) in
1299 the batch to update. If the href property is empty, effectively making the URI reference to the collection itself, the representation is
1300 to be applied to all resources in the batch",
1301     "items": {
1302         "additionalProperties": true,
1303         "properties": {
1304             "href": {
1305                 "description": "URI of the target resource relative assuming the collection URI as anchor",
1306                 "format": "uri",
1307                 "maxLength": 256,
1308                 "type": "string"
1309             },
1310             "rep": {
1311                 "oneOf": [
1312                     {
1313                         "description": "The response payload from a single resource",
1314                         "type": "object"
1315                     },
1316                     {
1317                         "description": " The response payload from a collection (batch) resource",
1318                         "type": "array"
1319                     }
1320                 ]

```

```

1321     }
1322 },
1323 "required": [
1324     "href",
1325     "rep"
1326 ],
1327 "type": "object"
1328 },
1329 "minItems": 1,
1330 "type": "array"
1331 }
1332
1333 ,
1334 "slinks" :
1335 {
1336     "description": "All forms of links in a collection",
1337     "oneOf": [
1338     {
1339         "description": "A set (array) of simple or individual OIC Links. In addition to properties required for an OIC Link, the
1340 identifier for that link in this set is also required",
1341         "items": {
1342             "properties": {
1343                 "anchor": {
1344                     "description": "This is used to override the context URI e.g. override the URI of the containing collection",
1345                     "format": "uri",
1346                     "maxLength": 256,
1347                     "type": "string"
1348                 },
1349                 "di": {
1350                     "description": "Unique identifier for device (UUID)",
1351                     "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
1352                     "type": "string"
1353                 },
1354                 "eps": {
1355                     "description": "the Endpoint information of the target Resource",
1356                     "items": {
1357                         "properties": {
1358                             "ep": {
1359                                 "description": "URI with Transport Protocol Suites + Endpoint Locator as specified in 10.2.1",
1360                                 "format": "uri",
1361                                 "type": "string"
1362                             },
1363                             "pri": {
1364                                 "description": "The priority among multiple Endpoints as specified in 10.2.3",
1365                                 "minimum": 1,
1366                                 "type": "integer"

```

```

1367         }
1368     },
1369     "type": "object"
1370 },
1371 "type": "array"
1372 },
1373 "href": {
1374     "description": "This is the target URI, it can be specified as a Relative Reference or fully-qualified URI. Relative
1375 Reference should be used along with the di parameter to make it unique.",
1376     "format": "uri",
1377     "maxLength": 256,
1378     "type": "string"
1379 },
1380 "if": {
1381     "description": "The interface set supported by this resource",
1382     "items": {
1383         "enum": [
1384             "oic.if.baseline",
1385             "oic.if.ll",
1386             "oic.if.b",
1387             "oic.if.rw",
1388             "oic.if.r",
1389             "oic.if.a",
1390             "oic.if.s"
1391         ],
1392         "type": "string"
1393     },
1394     "minItems": 1,
1395     "type": "array"
1396 },
1397 "ins": {
1398     "description": "The instance identifier for this web link in an array of web links - used in collections",
1399     "oneOf": [
1400         {
1401             "description": "An ordinal number that is not repeated - must be unique in the collection context",
1402             "type": "integer"
1403         },
1404         {
1405             "description": "Any unique string including a URI",
1406             "format": "uri",
1407             "maxLength": 256,
1408             "type": "string"
1409         },
1410         {
1411             "description": "Unique identifier (UUID)",
1412             "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",

```



```

1413         "type": "string"
1414     }
1415 ]
1416 },
1417 "p": {
1418     "description": "Specifies the framework policies on the Resource referenced by the target URI",
1419     "properties": {
1420         "bm": {
1421             "description": "Specifies the framework policies on the Resource referenced by the target URI for e.g. observable
1422 and discoverable",
1423             "type": "integer"
1424         }
1425     },
1426     "required": [
1427         "bm"
1428     ],
1429     "type": "object"
1430 },
1431 "rel": {
1432     "description": "The relation of the target URI referenced by the link to the context URI",
1433     "oneOf": [
1434         {
1435             "default": [
1436                 "hosts"
1437             ],
1438             "items": {
1439                 "maxLength": 64,
1440                 "type": "string"
1441             },
1442             "minItems": 1,
1443             "type": "array"
1444         },
1445         {
1446             "default": "hosts",
1447             "maxLength": 64,
1448             "type": "string"
1449         }
1450     ]
1451 },
1452 "rt": {
1453     "description": "Resource Type",
1454     "items": {
1455         "maxLength": 64,
1456         "type": "string"
1457     },
1458     "minItems": 1,

```

```

1459         "type": "array"
1460     },
1461     "title": {
1462         "description": "A title for the link relation. Can be used by the UI to provide a context",
1463         "maxLength": 64,
1464         "type": "string"
1465     },
1466     "type": {
1467         "default": "application/cbor",
1468         "description": "A hint at the representation of the resource referenced by the target URI. This represents the media
1469 types that are used for both accepting and emitting",
1470         "items": {
1471             "maxLength": 64,
1472             "type": "string"
1473         },
1474         "minItems": 1,
1475         "type": "array"
1476     }
1477 },
1478 "required": [
1479     "href",
1480     "rt",
1481     "if"
1482 ],
1483 "type": "object"
1484 },
1485 "type": "array"
1486 }
1487 ]
1488 }
1489
1490 ,
1491 "EasySetup" :
1492 {
1493     "description": "A collection is a set (array) of tagged-link or set (array) of simple links along with additional properties to
1494 describe the collection itself",
1495     "properties": {
1496         "cn": {
1497             "description": "Indicates an array of connection types that trigger an attempt to connect to the Enroller to start.",
1498             "items": {
1499                 "description": "Connection type to attempt. (1 : Wi-Fi, 2 : other entities / transports to be added in future (e.g. Connect
1500 to cloud / BLE))",
1501                 "type": "integer"
1502             },
1503             "type": "array"
1504         },

```

```

1505     "di": {
1506         "description": "The device ID which is a UUIDv4 string; used for backward compatibility with Spec A definition of
1507 /oic/res",
1508         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
1509         "type": "string"
1510     },
1511     "drel": {
1512         "description": "When specified this is the default relationship to use when an OIC Link does not specify an explicit
1513 relationship with *rel* parameter",
1514         "type": "string"
1515     },
1516     "id": {
1517         "anyOf": [
1518             {
1519                 "description": "A number that is unique to that collection; like an ordinal number that is not repeated",
1520                 "type": "integer"
1521             },
1522             {
1523                 "description": "A unique string that could be a hash or similarly unique",
1524                 "type": "string"
1525             },
1526             {
1527                 "description": "A unique string that could be a UUIDv4",
1528                 "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
1529                 "type": "string"
1530             }
1531         ],
1532         "description": "ID for the collection. Can be an value that is unique to the use context or a UUIDv4"
1533     },
1534     "lec": {
1535         "description": "Indicates a failure reason (0: NO error, 1: A given SSID is not found, 2: Wi-Fi's password is wrong, 3: IP
1536 address is not allocated, 4: No internet connection, 5: Timeout, 6: Wi-Fi Auth Type is not supported by the Enrollee, 7: Wi-Fi
1537 Encryption Type is not supported by the Enrollee, 8: Wi-Fi Auth Type is wrong (failure while connecting to the Enroller), 9: Wi-Fi
1538 Encryption Type is wrong (failure while connecting to the Enroller), 10~254: Reserved, 255: Unknown error)",
1539         "enum": [
1540             0,
1541             1,
1542             2,
1543             3,
1544             4,
1545             5,
1546             6,
1547             7,
1548             8,
1549             9,
1550             255

```

```

1551     ],
1552     "readOnly": true,
1553     "type": "integer"
1554   },
1555   "links": {
1556     "description": "All forms of links in a collection",
1557     "oneOf": [
1558       {
1559         "description": "A set (array) of simple or individual OIC Links. In addition to properties required for an OIC Link, the
1560 identifier for that link in this set is also required",
1561         "items": {
1562           "properties": {
1563             "anchor": {
1564               "description": "This is used to override the context URI e.g. override the URI of the containing collection",
1565               "format": "uri",
1566               "maxLength": 256,
1567               "type": "string"
1568             },
1569             "di": {
1570               "description": "Unique identifier for device (UUID)",
1571               "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
1572               "type": "string"
1573             },
1574             "eps": {
1575               "description": "the Endpoint information of the target Resource",
1576               "items": {
1577                 "properties": {
1578                   "ep": {
1579                     "description": "URI with Transport Protocol Suites + Endpoint Locator as specified in 10.2.1",
1580                     "format": "uri",
1581                     "type": "string"
1582                   },
1583                   "pri": {
1584                     "description": "The priority among multiple Endpoints as specified in 10.2.3",
1585                     "minimum": 1,
1586                     "type": "integer"
1587                   }
1588                 },
1589                 "type": "object"
1590               },
1591               "type": "array"
1592             },
1593             "href": {
1594               "description": "This is the target URI, it can be specified as a Relative Reference or fully-qualified URI. Relative
1595 Reference should be used along with the di parameter to make it unique.",
1596               "format": "uri",

```

```

1597         "maxLength": 256,
1598         "type": "string"
1599     },
1600     "if": {
1601         "description": "The interface set supported by this resource",
1602         "items": {
1603             "enum": [
1604                 "oic.if.baseline",
1605                 "oic.if.ll",
1606                 "oic.if.b",
1607                 "oic.if.rw",
1608                 "oic.if.r",
1609                 "oic.if.a",
1610                 "oic.if.s"
1611             ],
1612             "type": "string"
1613         },
1614         "minItems": 1,
1615         "type": "array"
1616     },
1617     "ins": {
1618         "description": "The instance identifier for this web link in an array of web links - used in collections",
1619         "oneOf": [
1620             {
1621                 "description": "An ordinal number that is not repeated - must be unique in the collection context",
1622                 "type": "integer"
1623             },
1624             {
1625                 "description": "Any unique string including a URI",
1626                 "format": "uri",
1627                 "maxLength": 256,
1628                 "type": "string"
1629             },
1630             {
1631                 "description": "Unique identifier (UUID)",
1632                 "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
1633                 "type": "string"
1634             }
1635         ]
1636     },
1637     "p": {
1638         "description": "Specifies the framework policies on the Resource referenced by the target URI",
1639         "properties": {
1640             "bm": {
1641                 "description": "Specifies the framework policies on the Resource referenced by the target URI for e.g.
1642                 observable and discoverable",

```

```

1643         "type": "integer"
1644     }
1645 },
1646 "required": [
1647     "bm"
1648 ],
1649 "type": "object"
1650 },
1651 "rel": {
1652     "description": "The relation of the target URI referenced by the link to the context URI",
1653     "oneOf": [
1654         {
1655             "default": [
1656                 "hosts"
1657             ],
1658             "items": {
1659                 "maxLength": 64,
1660                 "type": "string"
1661             },
1662             "minItems": 1,
1663             "type": "array"
1664         },
1665         {
1666             "default": "hosts",
1667             "maxLength": 64,
1668             "type": "string"
1669         }
1670     ]
1671 },
1672 "rt": {
1673     "description": "Resource Type",
1674     "items": {
1675         "maxLength": 64,
1676         "type": "string"
1677     },
1678     "minItems": 1,
1679     "type": "array"
1680 },
1681 "title": {
1682     "description": "A title for the link relation. Can be used by the UI to provide a context",
1683     "maxLength": 64,
1684     "type": "string"
1685 },
1686 "type": {
1687     "default": "application/cbor",
1688     "description": "A hint at the representation of the resource referenced by the target URI. This represents the

```

```

1689 media types that are used for both accepting and emitting",
1690     "items": {
1691         "maxLength": 64,
1692         "type": "string"
1693     },
1694     "minItems": 1,
1695     "type": "array"
1696 }
1697 },
1698 "required": [
1699     "href",
1700     "rt",
1701     "if"
1702 ],
1703 "type": "object"
1704 },
1705 "type": "array"
1706 }
1707 ]
1708 },
1709 "ps": {
1710     "description": "Indicates the easy setup status of the device. (0: Need to Setup, 1: Connecting to Enroller, 2: Connected to
1711 Enroller, 3: Failed to Connect to Enroller, 4~254: Reserved, 255: EOF)",
1712     "enum": [
1713         0,
1714         1,
1715         2,
1716         3
1717     ],
1718     "readOnly": true,
1719     "type": "integer"
1720 },
1721 "rt": {
1722     "items": {
1723         "enum": [
1724             "oic.r.easysetup",
1725             "oic.wk.col"
1726         ]
1727     },
1728     "maxItems": 2,
1729     "minItems": 2,
1730     "type": "array",
1731     "uniqueItems": true
1732 },
1733 "rts": {
1734     "description": "Defines the list of allowable resource types (for Target and anchors) in links included in the collection; new

```

```

1735 links being created can only be from this list",
1736     "items": {
1737         "maxLength": 64,
1738         "type": "string"
1739     },
1740     "minItems": 1,
1741     "readOnly": true,
1742     "type": "array"
1743 }
1744 },
1745 "required": [
1746     "ps",
1747     "lec",
1748     "cn"
1749 ],
1750 "type": "object"
1751 }
1752
1753 }
1754 }

```

1755 B.2.5 Property Definition

Property 명칭	값 유형	필수	엑세스 모드	설명
rep	복수 타입: schema 참조	예		
href	스트링	예		collection URI 를 앵커로 가정하여 상대적인 대상 resource 의 URI 를 나타낸다.
cn	배열: schema 참조	예		Enroller 에의 연결 시도를 시작하도록 하는 연결 유형의 배열을 나타낸다.
di	스트링			UUIDv4 스트링인 device ID; 시방서와의 역호환에 사용된다. /oic/res 의 정의.

Property 명칭	값 유형	필수	액세스 모드	설명
lec	정수	예	Read Only	연결 실패의 원인을 나타낸다 (0: 에러 없음, 1: 주어진 SSID 를 찾을 수 없다, 2: Wi-Fi 의 비밀번호 불일치, 3: IP 주소가 할당되어 있지 않다, 4: 인터넷에 연결되어 있지 않다, 5: 시간 초과, 6: Enrollee 에서 Wi-Fi Auth Type 을 지원하지 않는다, 7: Enrollee 에서 Wi-Fi Encryption Type 을 지원하지 않는다, 8: Wi-Fi Auth Type 이 맞지 않는다 (Enroller 에 연결 중의 실패), 9: Wi-Fi Encryption Type 이 맞지 않는다 (Enroller 에 연결 중의 실패), 10~254: Reserved, 255: 불명확한 에러)
drel	스트링			지정되면, OIC Link 가 *rel* 파라미터와의 명백한 관계를 지정하지 않을 때 사용하는 기본 관계를 나타낸다.
id	복수 타입: schema 참조			Collection 을 식별하기 위한 ID. 유스 컨텍스트 또는 UUIDv4 에 고유한 값일 수 있다.
rt	배열: schema 참조			
rts	배열: schema 참조		Read Only	collection 에 포함된 링크에서 (대상 또는 앵커에 대해) 허용 가능한 resource type 의 목록을 정의한다; 신규 링크는 이 목록으로부터만 생성할 수 있다.
ps	정수	예	Read Only	device 의 easy setup 상태를 나타낸다. (0: Setup 필요, 1: Enroller 에 연결 중, 2: Enroller 에 연결됨, 3: Enroller 에 연결 실패, 4~254: Reserved, 255: EOF)
links	복수 타입: schema 참조			collection 내의 모든 링크의 형태.

Property 명칭	값 유형	필수	액세스 모드	설명
anchor	스트링			컨텍스트 URI 를 대체하는데 사용된다. 예를 들어, 포함하는 collection 의 URI 를 대체한다.
di	스트링			device 의 고유 식별자 (UUID)
title	스트링			링크 관계의 타이틀. 컨텍스트 제공을 위해 UI 에 의해 사용될 수 있다.
p	객체: schema 참조			대상 URI 에 의해 참조되는 Resource 에 관한 프레임워크 정책을 지정한다.
rel	복수 타입: schema 참조			링크에 의해 참조되는 대상 URI 의 컨텍스트 URI 에 대한 관계.
rt	배열: schema 참조	예		Resource Type
ins	복수 타입: schema 참조			웹 링크의 배열에서 해당 웹 링크에 대한 인스턴스 식별자 - collections 에서 사용됨.
eps	배열: schema 참조			대상 Resource 의 Endpoint 정보.
if	배열: schema 참조	예		해당 resource 에서 지원하는 interface set.
type	배열: schema 참조			대상 URI 에 의해 참조되는 resource 의 표현에 대한 힌트. 수신 및 발신에 사용되는 미디어 유형을 나타낸다.
href	스트링	예		대상 URI. 상대적 참조 또는 전체 URI 로 지정할 수 있다. 상대적 참조는 di 파라미터와 함께 사용함으로써 고유하게 될 수 있다.
rep	복수 타입: schema 참조	예		
href	스트링	예		collection URI 를 앵커로 가정하여 상대적인 대상 resource 의 URI 를 나타낸다.

1756

B.2.6 CRUDN behaviour

Resource	Create	Read	Update	Delete	Notify
/example/EasySetupBaselineInterfaceResURI		get			

1757 **B.3 Wi-Fi Configuration Resource**

1758 **B.3.1 개요**

1759 WiFiConf resource 는 개봉하는 device 가 기존의 Wi-Fi AP 에 연결하기 위한 필수 정보를 저장한다.

1760 WiFiConf resource 의 property 를 검색한다.

1761 정보는 다음을 포함한다.

1762 1. Wi-Fi SSID 및 비밀번호

1763 2. Wi-Fi Security 유형 (즉, auth type 및 encryption type)

1764 3. Wi-Fi 하드웨어 성능 (즉, 지원되는 주파수, 모드, auth type 및 encryption type)

1765

1766 **B.3.2 잘 알려진 URI**

1767 /example/WiFiConfBaselineInterfaceResURI

1768 **B.3.3 Resource Type**

1769 resource type (rt)는 ['oic.r.wificonf']로 정의된다.

1770 **B.3.4 Swagger2.0 정의**

```
1771 {
1772   "swagger": "2.0",
1773   "info": {
1774     "title": "Wi-Fi Configuration Resource Baseline Interface",
1775     "version": "v0.0.3-20170611",
1776     "license": {
1777       "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
1778       "x-description": "Redistribution and use in source and binary forms, with or without modification, are permitted provided that
1779 the following conditions are met:\n      1. Redistributions of source code must retain the above copyright notice, this list of
1780 conditions and the following disclaimer.\n      2. Redistributions in binary form must reproduce the above copyright notice, this
1781 list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open Connectivity Foundation, INC. \n\n      AS IS\n\n      AND ANY EXPRESS OR IMPLIED
1782 WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
1783 PARTICULAR PURPOSE OR WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n\n      IN NO EVENT SHALL THE Open
1784 Connectivity Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
1785 CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
1786 USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n\n      HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
1787 WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
1788 THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.\n\n"
1789     }
1790   },
1791   "schemes": ["http"],
1792   "consumes": ["application/json"],
```

```

1794     "produces": ["application/json"],
1795     "paths": {
1796         "/example/WiFiConfBaselineInterfaceResURI" : {
1797             "get": {
1798                 "description": "WiFiConf resource stores essential information to help an unboxing deviceWn to connect to an existing Wi-Fi
1799 AP.Wn Retrieve properties of WiFiConf resource.Wn The information includes :Wn1. Wi-Fi SSID and passwordWn2. Wi-Fi Security type
1800 (i.e. auth type and encryption type)Wn3. Wi-Fi hardware capability (i.e. supported frequencies, modes,Wn auth types and encryption
1801 types)Wn",
1802                 "parameters": [
1803                     {"$ref": "#/parameters/interface-baseline"}
1804                 ],
1805                 "responses": {
1806                     "200": {
1807                         "description": "",
1808                         "x-example":
1809                             {
1810                                 "rt": ["oic.r.wificonf"],
1811                                 "swmt": ["A", "B", "G"],
1812                                 "swf": ["2.4G", "5G"],
1813                                 "tnn": "Home_AP_SSID",
1814                                 "cd": "Home_AP_PWD",
1815                                 "wat": "WPA2_PSK",
1816                                 "wet": "TKIP",
1817                                 "swat": ["WPA_PSK", "WPA2_PSK"],
1818                                 "swet": ["TKIP", "AES", "TKIP_AES"]
1819                             }
1820                         ,
1821                         "schema": { "$ref": "#/definitions/WiFiConf" }
1822                     }
1823                 }
1824             },
1825             "post": {
1826                 "description": "Deliver Wi-Fi AP's information for an unboxing device to connect to it.Wn",
1827                 "parameters": [
1828                     {"$ref": "#/parameters/interface-baseline"},
1829                     {
1830                         "name": "body",
1831                         "in": "body",
1832                         "required": true,
1833                         "schema": { "$ref": "#/definitions/WiFiConfUpdate" },
1834                         "x-example":
1835                             {
1836                                 "tnn": "Home_AP_SSID",
1837                                 "cd": "Home_AP_PWD",
1838                                 "wat": "WPA2_PSK",
1839                                 "wet": "AES"

```

```

1840     }
1841 }
1842 ],
1843 "responses": {
1844     "200": {
1845         "description": "",
1846         "x-example":
1847         {
1848             "tnn": "Home_AP_SSID",
1849             "cd": "Home_AP_PWD",
1850             "wat": "WPA2_PSK",
1851             "wet": "AES"
1852         }
1853     },
1854     "schema": { "$ref": "#/definitions/WiFiConfUpdate" }
1855 }
1856 }
1857 }
1858 },
1859 "/example/WiFiConfRWInterfaceResURI" : {
1860     "get": {
1861         "description": "WiFiConf resource stores essential information to help an unboxing deviceWn
1862 AP.WnRetrieve properties of WiFiConf resource that can be updated by a client.Wn",
1863         "parameters": [
1864             {"$ref": "#/parameters/interface-rw"}
1865         ],
1866         "responses": {
1867             "200": {
1868                 "description": "",
1869                 "x-example":
1870                 {
1871                     "tnn": "Home_AP_SSID",
1872                     "cd": "Home_AP_PWD",
1873                     "wat": "WPA2_PSK",
1874                     "wet": "AES"
1875                 }
1876             },
1877             "schema": { "$ref": "#/definitions/WiFiConfUpdate" }
1878         }
1879     }
1880 },
1881 "post": {
1882     "description": "Deliver Wi-Fi AP's information for an unboxing device to connect to it.Wn",
1883     "parameters": [
1884         {"$ref": "#/parameters/interface-rw"},
1885         {

```

```

1886     "name": "body",
1887     "in": "body",
1888     "required": true,
1889     "schema": { "$ref": "#/definitions/WiFiConfUpdate" },
1890     "x-example":
1891     {
1892         "tnn": "Home_AP_SSID",
1893         "cd": "Home_AP_PWD",
1894         "wat": "WPA2_PSK",
1895         "wet": "AES"
1896     }
1897 }
1898 ],
1899 "responses": {
1900     "200": {
1901         "description": "",
1902         "x-example":
1903         {
1904             "tnn": "Home_AP_SSID",
1905             "cd": "Home_AP_PWD",
1906             "wat": "WPA2_PSK",
1907             "wet": "AES"
1908         }
1909     },
1910     "schema": { "$ref": "#/definitions/WiFiConfUpdate" }
1911 }
1912 }
1913 }
1914 }
1915 },
1916 "parameters": {
1917     "interface-rw" : {
1918         "in" : "query",
1919         "name" : "if",
1920         "type" : "string",
1921         "enum" : ["oic.if.rw"]
1922     },
1923     "interface-baseline" : {
1924         "in" : "query",
1925         "name" : "if",
1926         "type" : "string",
1927         "enum" : ["oic.if.baseline"]
1928     },
1929     "interface-all" : {
1930         "in" : "query",
1931         "name" : "if",

```

```

1932     "type" : "string",
1933     "enum" : ["oic.if.baseline", "oic.if.rw"]
1934 }
1935 },
1936 "definitions": {
1937     "WiFiConf" :
1938     {
1939         "properties": {
1940             "cd": {
1941                 "description": "Indicates credential information of Wi-Fi AP",
1942                 "pattern": "^.*$",
1943                 "type": "string"
1944             },
1945             "swat": {
1946                 "description": "Indicates supported Wi-Fi Auth types. It can be multiple",
1947                 "items": {
1948                     "description": "Indicates Wi-Fi Auth Type",
1949                     "enum": [
1950                         "None",
1951                         "WEP",
1952                         "WPA_PSK",
1953                         "WPA2_PSK"
1954                     ],
1955                     "type": "string"
1956                 },
1957                 "readOnly": true,
1958                 "type": "array"
1959             },
1960             "swet": {
1961                 "description": "Indicates supported Wi-Fi Encryption types. It can be multiple",
1962                 "items": {
1963                     "description": "Indicates Wi-Fi Encryption Type",
1964                     "enum": [
1965                         "None",
1966                         "WEP_64",
1967                         "WEP_128",
1968                         "TKIP",
1969                         "AES",
1970                         "TKIP_AES"
1971                     ],
1972                     "type": "string"
1973                 },
1974                 "readOnly": true,
1975                 "type": "array"
1976             },
1977             "swf": {

```

```

1978     "description": "Indicates Supported Wi-Fi frequencies by the Enrollee. Can be multiple. Valid values are ('2.4G', '5G')",
1979     "items": {
1980         "pattern": "^(2\\W\\W.4|5)G$",
1981         "type": "string"
1982     },
1983     "readOnly": true,
1984     "type": "array"
1985 },
1986 "swmt": {
1987     "description": "Indicates supported Wi-Fi mode types. It can be multiple",
1988     "items": {
1989         "description": "Supported Wi-Fi Mode Type.",
1990         "enum": [
1991             "A",
1992             "B",
1993             "G",
1994             "N",
1995             "AC"
1996         ],
1997         "type": "string"
1998     },
1999     "readOnly": true,
2000     "type": "array"
2001 },
2002 "tnn": {
2003     "description": "Indicates Target Network Name (SSID of Wi-Fi AP)",
2004     "pattern": "^[^.*$]",
2005     "type": "string"
2006 },
2007 "wat": {
2008     "description": "Indicates Wi-Fi Auth Type",
2009     "enum": [
2010         "None",
2011         "WEP",
2012         "WPA_PSK",
2013         "WPA2_PSK"
2014     ],
2015     "type": "string"
2016 },
2017 "wet": {
2018     "description": "Indicates Wi-Fi Encryption Type",
2019     "enum": [
2020         "None",
2021         "WEP_64",
2022         "WEP_128",
2023         "TKIP",

```



```

2024         "AES",
2025         "TKIP_AES"
2026     ],
2027     "type": "string"
2028 }
2029 },
2030 "required": [
2031     "swmt",
2032     "swf",
2033     "swat",
2034     "swet",
2035     "tnn",
2036     "wat",
2037     "wet"
2038 ],
2039 "type": "object"
2040 }
2041
2042 ,
2043 "WiFiConfUpdate" :
2044     {
2045     "properties": {
2046         "cd": {
2047             "description": "Indicates credential information of Wi-Fi AP",
2048             "pattern": "^.*$",
2049             "type": "string"
2050         },
2051         "tnn": {
2052             "description": "Indicates Target Network Name (SSID of Wi-Fi AP)",
2053             "pattern": "^.*$",
2054             "type": "string"
2055         },
2056         "wat": {
2057             "description": "Indicates Wi-Fi Auth Type",
2058             "enum": [
2059                 "None",
2060                 "WEP",
2061                 "WPA_PSK",
2062                 "WPA2_PSK"
2063             ]
2064         },
2065         "wet": {
2066             "description": "Indicates Wi-Fi Encryption Type",
2067             "enum": [
2068                 "None",
2069                 "WEP_64",

```

```

2070         "WEP_128",
2071         "TKIP",
2072         "AES",
2073         "TKIP_AES"
2074     ]
2075 }
2076 },
2077 "required": [
2078     "tnn",
2079     "wat",
2080     "wet"
2081 ],
2082 "type": "object"
2083 }
2084
2085 }
2086 }

```

2087 B.3.5 Property 정의

Property 명칭	값 유형	필수	액세스 모드	설명
swet	배열: schema 참조	예	Read Only	지원되는 Wi-Fi Encryption type 을 나타낸다. 복수의 유형일 수 있다.
wet	스트링	예		Wi-Fi Encryption Type 을 나타낸다.
swmt	배열: schema 참조	예	Read Only	지원되는 Wi-Fi mode type 을 나타낸다. 복수의 유형일 수 있다.
swat	배열: schema 참조	예	Read Only	지원되는 Wi-Fi Auth type 을 나타낸다. 복수의 유형일 수 있다.
cd	스트링			Wi-Fi AP 의 크리덴셜 정보를 나타낸다.
tnn	스트링	예		대상 네트워크 명칭을 나타낸다 (Wi-Fi AP 의 SSID)
wat	스트링	예		Wi-Fi Auth Type 을 나타낸다.

Property 명칭	값 유형	필수	액세스 모드	설명
swf	배열: schema 참조	예	Read Only	Enrollee 에서 지원하는 Wi-Fi 주파수를 나타낸다. 복수의 주파수일 수 있다. 유효한 값은 ('2.4G', '5G')이다.
tnn	스트링	예		대상 네트워크 명칭을 나타낸다 (Wi-Fi AP 의 SSID)
wet	복수 타입: schema 참조	예		Wi-Fi Encryption Type 을 나타낸다.
wat	복수 타입: schema 참조	예		Wi-Fi Auth Type 을 나타낸다.
cd	스트링			Wi-Fi AP 의 크리덴셜 정보를 나타낸다.

2088 B.3.6 CRUDN 동작

Resource	Create	Read	Update	Delete	Notify
/example/WiFiConfBaselineInterfaceResURI		get	post		

2089