

1 **OCF “Essen” – oic.wk.rd is only to be used by a Cloud – Core Technology WG CR 2851**

2  
3

4 Legal Disclaimer

5

6 THIS IS A DRAFT SPECIFICATION DOCUMENT ONLY AND HAS NOT BEEN ADOPTED BY THE  
7 OPEN CONNECTIVITY FOUNDATION. THIS DRAFT DOCUMENT MAY NOT BE RELIED UPON  
8 FOR ANY PURPOSE OTHER THAN REVIEW OF THE CURRENT STATE OF THE DEVELOPMENT  
9 OF THIS DRAFT DOCUMENT. THE OPEN CONNECTIVITY FOUNDATION AND ITS MEMBERS  
10 RESERVE THE RIGHT WITHOUT NOTICE TO YOU TO CHANGE ANY OR ALL PORTIONS  
11 HEREOF, DELETE PORTIONS HEREOF, MAKE ADDITIONS HERETO, DISCARD THIS DRAFT  
12 DOCUMENT IN ITS ENTIRETY OR OTHERWISE MODIFY THIS DRAFT DOCUMENT AT ANY  
13 TIME. YOU SHOULD NOT AND MAY NOT RELY UPON THIS DRAFT DOCUMENT IN ANY WAY,  
14 INCLUDING BUT NOT LIMITED TO THE DEVELOPMENT OF ANY PRODUCTS OR SERVICES.  
15 IMPLEMENTATION OF THIS DRAFT DOCUMENT IS DONE AT YOUR OWN RISK AMEND AND  
16 IT IS NOT SUBJECT TO ANY LICENSING GRANTS OR COMMITMENTS UNDER THE OPEN  
17 CONNECTIVITY FOUNDATION INTELLECTUAL PROPERTY RIGHTS POLICY OR OTHERWISE.  
18 IN CONSIDERATION OF THE OPEN CONNECTIVITY FOUNDATION GRANTING YOU ACCESS  
19 TO THIS DRAFT DOCUMENT, YOU DO HEREBY WAIVE ANY AND ALL CLAIMS ASSOCIATED  
20 HERewith INCLUDING BUT NOT LIMITED TO THOSE CLAIMS DISCUSSED BELOW, AS WELL  
21 AS CLAIMS OF DETRIMENTAL RELIANCE.

22 The OCF logo is a trademark of Open Connectivity Foundation, Inc. in the United States or other  
23 countries. \*Other names and brands may be claimed as the property of others.

24 Copyright © 2019 Open Connectivity Foundation, Inc. All rights reserved.

25 Copying or other form of reproduction and/or distribution of these works are strictly prohibited.

26



DRAFT

## OCF Cloud Specification Essen

Open Connectivity Foundation (OCF)  
admin@openconnectivity.org

DRAFT

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

**CONTENTS**

1	Scope	1
2	Normative references	1
3	Terms, definitions, and abbreviated terms	2
3.1	Terms and definitions	2
3.2	Abbreviated terms	2
4	Document conventions and organization	3
4.1	Conventions	3
4.2	Notation	3
5	Overview	5
5.1	Introduction	5
5.2	Architecture	5
5.3	Interaction Flow	6
5.4	Cloud Operational Flow	7
5.4.1	Pre-requisites and OCF Cloud User Account Creation	8
5.4.2	Mediator registration with the OCF Cloud	8
5.4.3	Device provisioning by the Mediator	8
5.4.4	Device Registration with the OCF Cloud	9
5.4.5	Connection with the OCF Cloud	9
5.4.6	Publishing Links to the OCF Cloud RD	9
5.4.7	Client to Server communication through the OCF Cloud	9
5.4.8	Refreshing connection with the OCF Cloud	10
5.4.9	Closing connection with the OCF Cloud	10
5.4.10	Deregistering from the OCF Cloud	10
6	Resource model	12
6.1	OCF Cloud Resource Directory	12
6.1.1	Indirect discovery for lookup of the Resources	12
6.1.2	Resource Directory Definition	12
6.1.3	RD operational flows	14
6.2	CoAPCloudConf Resource	18
6.2.1	Introduction	18
6.2.2	Resource Definition	19
6.2.3	Error Handling	20
7	Network and connectivity	21
8	Functional interactions	22
8.1	Onboarding, Provisioning, and Configuration	22
8.1.1	Overview	22
8.1.2	Use of Mediator	22
8.1.3	Device Connection to the OCF Cloud	25

70	8.1.4	Device Registration with the OCF Cloud .....	25
71	8.2	Resource Publication .....	26
72	8.3	Client Registration with the OCF Cloud .....	27
73	8.4	Resource Discovery .....	27
74	8.5	Device Deregistration from the OCF Cloud.....	29
75	9	Security .....	29
76	Annex A (normative)	Swagger2.0 definitions .....	30
77	A.1	List of Resource Type definitions .....	30
78	A.2	Resource directory resource .....	30
79	A.2.1	Introduction .....	30
80	A.2.2	Well-known URI .....	30
81	A.2.3	Resource type .....	30
82	A.2.4	OpenAPI 2.0 definition.....	30
83	A.2.5	Property definition .....	35
84	A.2.6	CRUDN behaviour .....	35
85	A.3	CoAP Cloud Configuration Resource .....	36
86	A.3.1	Introduction .....	36
87	A.3.2	Example URI .....	36
88	A.3.3	Resource type .....	36
89	A.3.4	OpenAPI 2.0 definition.....	36
90	A.3.5	Property definition .....	39
91	A.3.6	CRUDN behaviour .....	40
92			
93			

94	
95	
96	
97	Figure 1 – OCF Cloud Architecture ..... 6
98	Figure 2 – OCF Cloud interaction model ..... 7
99	Figure 3 – Overall Operational State Machine ..... 12
100	Figure 4 – Indirect discovery of Resources by via an RD ..... 12
101	Figure 5 – RD discovery and RD supported query of Resources support..... 14
102	Figure 7 – Registration with OCF Cloud ..... 22
103	Figure 8 – Device Provisioning by the Mediator ..... 25
104	Figure 9 – Resource publication to the OCF Cloud..... 27
105	Figure 10 – Resource discovery through OCF Cloud..... 28
106	Figure 11 – Request routing through OCF Cloud..... 29
107	
108	

## Tables

109	
110	
111	Table 1 – OCF Cloud Interaction Flow .....7
112	Table 2 – "oic.wk.rd" Resource Type definition ..... 13
113	Table 3 – "oic.wk.rd" Properties ..... 13
114	Table 4 – CoAPCloudConf Resource ..... 19
115	Table 5 – oic.r.coapcloudconf Resource Type definition.....20
116	Table 6 – Device to OCF Cloud Registration Flow.....22
117	Table 7 – Device Provisioning by the Mediator.....25
118	Table A.1 – Alphabetized list of resources ..... 30
119	Table D-22 – The Property definitions of the Resource with type "rt" = "oic.wk.rd". .....35
120	Table D-23 – The CRUDN operations of the Resource with type "rt" = "oic.wk.rd".....35
121	Table A.2 – The Property definitions of the Resource with type "rt" = "oic.r.coapcloudconf". .39
122	Table A.3 – The CRUDN operations of the Resource with type "rt" = "oic.r.coapcloudconf"...40
123	

DRAFT

124 **1 Scope**

125 This document defines functional extensions to the capabilities defined in ISO/IEC 30118-1:2018  
126 to meet the requirements of the OCF Cloud. This document specifies new Resource Types to  
127 enable the functionality and any extensions to the existing capabilities defined in ISO/IEC 30118-  
128 1:2018.

129 **2 Normative references**

130 The following documents are referred to in the text in such a way that some or all of their content  
131 constitutes requirements of this document. For dated references, only the edition cited applies. For  
132 undated references, the latest edition of the referenced document (including any amendments)  
133 applies.

134 ISO/IEC 30118-1:2018 *Information technology -- Open Connectivity Foundation (OCF)*  
135 *Specification -- Part 1: Core specification*  
136 <https://www.iso.org/standard/53238.html>  
137 Latest version available at: [https://openconnectivity.org/specs/OCF\\_Core\\_Specification.pdf](https://openconnectivity.org/specs/OCF_Core_Specification.pdf)

138 ISO/IEC 30118-2:2018 *Information technology -- Open Connectivity Foundation (OCF)*  
139 *Specification -- Part 2: Security specification*  
140 <https://www.iso.org/standard/74239.html>  
141 Latest version available at: [https://openconnectivity.org/specs/OCF\\_Security\\_Specification.pdf](https://openconnectivity.org/specs/OCF_Security_Specification.pdf)

142 OCF Wi-Fi Easy Setup, *Open Connectivity Foundation Wi-Fi Easy Setup, Version 2.0.1*  
143 Latest version available at:  
144 [https://openconnectivity.org/specs/OCF\\_Wi-Fi\\_Easy\\_Setup\\_Specification.pdf](https://openconnectivity.org/specs/OCF_Wi-Fi_Easy_Setup_Specification.pdf)

145 IETF RFC 6749, *The OAuth 2.0 Authorization Framework*, October 2012  
146 <https://tools.ietf.org/html/rfc6749>

147 IETF RFC 6750, *The OAuth 2.0 Authorization Framework: Bearer Token Usage*, October 2012  
148 <https://tools.ietf.org/html/rfc6750>

149 IETF RFC 8323, *CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets*,  
150 February 2018  
151 <https://tools.ietf.org/html/rfc8323>

152 OpenAPI specification, *fka Swagger RESTful API Documentation Specification*, Version 2.0  
153 <https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md>

154



155 **3 Terms, definitions, and abbreviated terms**

156 **3.1 Terms and definitions**

157 For the purposes of this document, the terms and definitions given in ISO/IEC 30118-1:2018 and  
158 ISO/IEC 30118-2:2018 and the following apply.

159 ISO and IEC maintain terminological databases for use in standardization at the following  
160 addresses:

- 161 – ISO Online browsing platform: available at <https://www.iso.org/obp>
- 162 – IEC Electropedia: available at <http://www.electropedia.org/>

163 **3.1.1**

164 **Cloud Provider**

165 entity or organization that hosts an OCF Cloud (3.1.2).

166 **3.1.2**

167 **OCF Cloud**

168 an OCF Cloud is not an OCF Device, but a logical entity that is owned by the Cloud Provider (3.1.1).  
169 An OCF Cloud is authorised to communicate with a Device on behalf of the OCF Cloud User.

170 **3.1.3**

171 **Resource Directory**

172 a set of descriptions of Resources where the actual Resources are held on Servers external to the  
173 entity hosting the Resource Directory (3.1.3), allowing lookups to be performed for those Resources

174 **3.2 Abbreviated terms**

175 **3.2.1**

176 **UX**

177 User Experience

178

## 179 **4 Document conventions and organization**

### 180 **4.1 Conventions**

181 In this document a number of terms, conditions, mechanisms, sequences, parameters, events,  
182 states, or similar terms are printed with the first letter of each word in uppercase and the rest  
183 lowercase (e.g., Network Architecture). Any lowercase uses of these words have the normal  
184 technical English meaning.

### 185 **4.2 Notation**

186 In this document, features are described as required, recommended, allowed or DEPRECATED as  
187 follows:

188 Required (or shall or mandatory)(M).

- 189 – These basic features shall be implemented to comply with Core Architecture. The phrases "shall  
190 not", and "PROHIBITED" indicate behaviour that is prohibited, i.e. that if performed means the  
191 implementation is not in compliance.

192 Recommended (or should)(S).

- 193 – These features add functionality supported by Core Architecture and should be implemented.  
194 Recommended features take advantage of the capabilities Core Architecture, usually without  
195 imposing major increase of complexity. Notice that for compliance testing, if a recommended  
196 feature is implemented, it shall meet the specified requirements to be in compliance with these  
197 guidelines. Some recommended features could become requirements in the future. The phrase  
198 "should not" indicates behaviour that is permitted but not recommended.

199 Allowed (may or allowed)(O).

- 200 – These features are neither required nor recommended by Core Architecture, but if the feature  
201 is implemented, it shall meet the specified requirements to be in compliance with these  
202 guidelines.

203 DEPRECATED.

- 204 – Although these features are still described in this document, they should not be implemented  
205 except for backward compatibility. The occurrence of a deprecated feature during operation of  
206 an implementation compliant with the current document has no effect on the implementation's  
207 operation and does not produce any error conditions. Backward compatibility may require that  
208 a feature is implemented and functions as specified but it shall never be used by  
209 implementations compliant with this document.

210 Conditionally allowed (CA)

- 211 – The definition or behaviour depends on a condition. If the specified condition is met, then the  
212 definition or behaviour is allowed, otherwise it is not allowed.

213 Conditionally required (CR)

- 214 – The definition or behaviour depends on a condition. If the specified condition is met, then the  
215 definition or behaviour is required. Otherwise the definition or behaviour is allowed as default  
216 unless specifically defined as not allowed.

217

218 Strings that are to be taken literally are enclosed in "double quotes".

219 Words that are emphasized are printed in *italic*.

DRAFT

220 **5 Overview**

221 **5.1 Introduction**

222 An OCF Cloud extends the use of CoAP to enable a Device to interact with a cloud by utilizing  
223 following features

- 224 – CoAP over TCP protocol defined in ISO/IEC 30118-1:2018
- 225 – The requirements within this document including those for a Resource Directory
- 226 – Security requirements and SVRs defined within the ISO/IEC 30118-2:2018

227 Devices which are not within a single local network may interact with each other using CoAP over  
228 TCP (see ISO/IEC 30118-1:2018) via an OCF Cloud. At any point in time, a Device is configured  
229 to use at most one OCF Cloud. The OCF Cloud groups Devices that belong to same OCF Cloud  
230 User under an OCF Cloud created User ID. All the Devices registered to the OCF Cloud and  
231 belonging to the same User ID can communicate with each other subject to the Device(s)  
232 authorising the OCF Cloud in the ACE2 policies.

233 Annex A specifies the Resource Type definitions using the schema defined in the  
234 OpenAPI specification as the API definition language that shall be followed by an OCF Device  
235 realizing the Resources specified in this document.

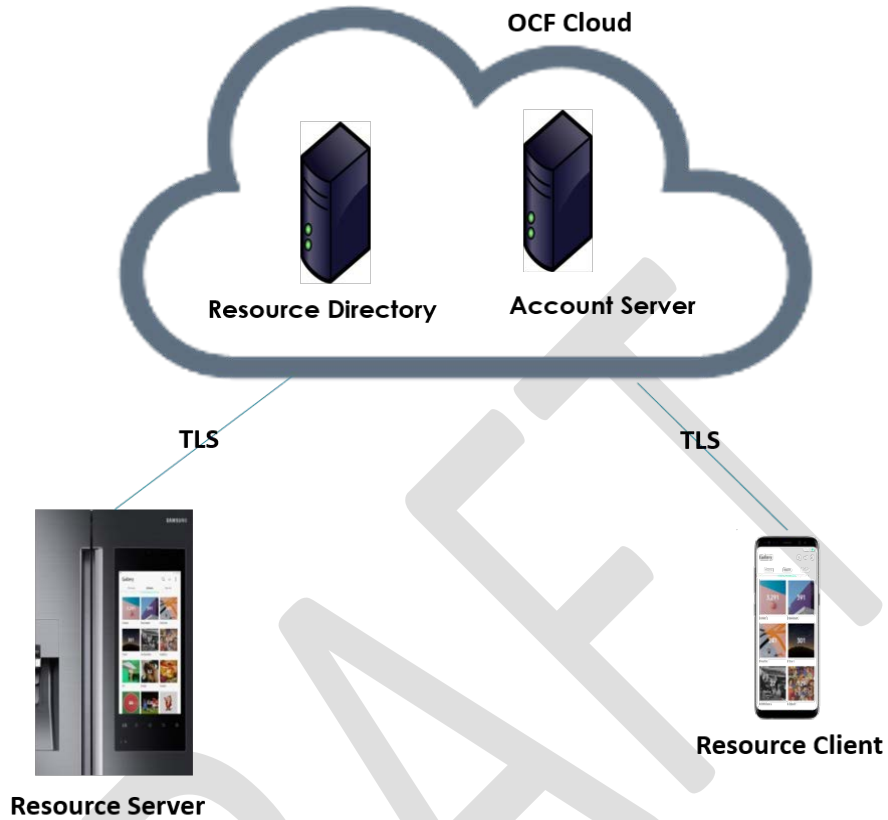
236 Note that an OCF Cloud is not an OCF Device, but a logical entity that is owned by the Cloud  
237 Provider. An OCF Cloud is authorized to communicate with a Device by the OCF Cloud User

238 **5.2 Architecture**

239 The OCF Cloud is a logical entity to which an OCF Device communicates via a persistent TLS  
240 connection. It encapsulates two functions:

- 241 – an account server function which is a logical entity that handles Device registration, Access  
242 Token validation and handles sign-in and token-refresh requests from the Device. An OCF  
243 Cloud User creates offline an account on the account server (by means of the mediator). The  
244 account server is then also used to register the Devices (Clients and Servers) per account.  
245 Note that all accounts are fully separated, e.g. logging into account A does not give access to  
246 Devices registered to account B.
- 247 – a Resource Directory as defined by this document. The Resource Directory exposes Resource  
248 information published by Devices. A Client, when discovering Devices, receives a response  
249 from the Resource Directory on behalf of the Device. With information included in the response  
250 from the Resource Directory, the Client may connect to the Device via the OCF Cloud.

251 This is illustrated in Figure 1.



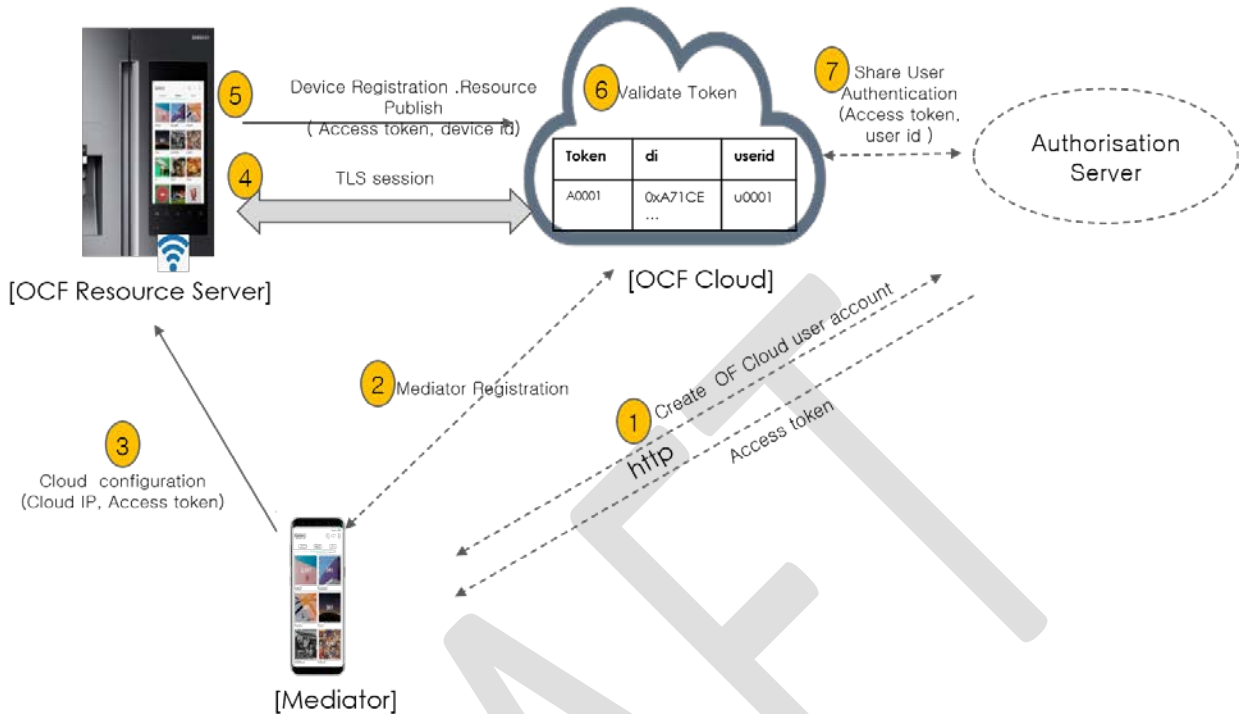
252

253

**Figure 1 – OCF Cloud Architecture**

### 254 **5.3 Interaction Flow**

255 This clause describes how the elements with the overall OCF Cloud interact. Figure 2 provides an  
256 overall introduction, Table 1 provides additional context to the elements in the flow.



**Figure 2 – OCF Cloud interaction model**

**Table 1 – OCF Cloud Interaction Flow**

Steps	Description
1	The Mediator obtains an Access Token for the OCF Cloud User from an Authorisation Provider
2	The Mediator registers with the OCF Cloud
3	The Mediator provisions "oic.r.coapcloudconf" on the Device with an Access Token, the URL of the OCF Cloud, the identity (UUID) of the OCF Cloud, and optionally an Authorisation Provider Name.
4, 5	The Device establishes a TLS session to the OCF Cloud and subsequently registers with the OCF Cloud
6, 7	The OCF Cloud validates the registration request and authorises the Access Token. Returning information to the Device in the "uid" of the OCF Cloud User and the expiration information of the Access Token.

261

262 In the case where the OCF Cloud also acts as the Authorisation Server step 1 from Table 1 may  
263 be between the Mediator and the OCF Cloud in which case step 7 is not required.

264 **5.4 Cloud Operational Flow**

265 The sub-clauses listed provide an informative overview of the flow which results on a Device being  
266 registered with an OCF Cloud and Client interaction with that Device. The clauses provide

267 references to the applicable clauses within this document and other documents that provide  
268 normative details.

269 The flow consists of the following high-level steps:

- 270 – Pre-requisites and OCF Cloud User account creation (see 5.4.1)
- 271 – Mediator registration with the OCF Cloud (see 5.4.2)
- 272 – Device provisioning by the Mediator (see 5.4.3)
- 273 – Device registration with the OCF Cloud (see 5.4.4)
- 274 – Device connection with the OCF Cloud (see 5.4.5)
- 275 – Devices Publishing Links to the OCF Cloud RD (see 5.4.6)
- 276 – Client to Server communication through the OCF Cloud (see 5.4.7)
- 277 – Device refreshing connection with the OCF Cloud (see 5.4.8)
- 278 – Device closing connection with the OCF Cloud (see 5.4.9)
- 279 – Device de-registering from the OCF Cloud (see 5.4.10)

#### 280 **5.4.1 Pre-requisites and OCF Cloud User Account Creation**

281 The OCF Cloud User has a Device that they want to hook up to the OCF Cloud so that they can  
282 access it remotely.

283 The Device is onboarded to the OCF Network as defined in ISO/IEC 30118-2:2018.

284 The OCF Cloud User makes use of a Mediator to provision the Device. A Mediator is a logical  
285 function that may be on the OCF Cloud User's personal device (e.g. phone) or elsewhere. The  
286 Mediator is configured with or through some out of band process to obtain the URL of the OCF  
287 Cloud (e.g. the Mediator may be an application from the Cloud Provider).

288 The OCF Cloud User has access credentials for authenticating the OCF Cloud User to the  
289 Authorisation Provider (i.e. user name/password or similar)

#### 290 **5.4.2 Mediator registration with the OCF Cloud**

291 See 8.1.2.2, 8.1.2.3.

292 Via some trigger (e.g. a UX or other out of bounds mechanism), the Mediator authenticates the  
293 OCF Cloud User to the Authorisation Provider and requests Access Token from an Authorisation  
294 Provider.

295 The Mediator registers by providing its Access Token to the OCF Cloud which verifies the token  
296 and creates a User ID with which the Mediator is associated. All instances of a Mediator for the  
297 same OCF Cloud User will be associated with the same User ID. Similarly, this same User ID may  
298 be used to assign multiple Devices to the same OCF Cloud User

#### 299 **5.4.3 Device provisioning by the Mediator**

300 See 8.1.2.3; see also ISO/IEC 30118-2:2018 clause 7.5.2

301 The Mediator connects to the Device through normal OCF processes. The Mediator then requests  
302 an Access Token from the OCF Cloud for the Device being provisioned. The Mediator updates the  
303 "oic.r.coapcloudconf" Resource on the Device with the Access Token received from the OCF Cloud,  
304 the OCF Cloud URI, and the OCF Cloud UUID. The Mediator may also provide the Auth Provider

305 Name. Note that this Access Token may only be used one time for the initial Device Registration  
306 with the OCF Cloud.

#### 307 **5.4.4 Device Registration with the OCF Cloud.**

308 See 8.1.3 and 8.1.4; see also ISO/IEC 30118-2:2018 clauses 10.5, 13.11, 13.12

309 On configuration of the "oic.r.coapcloudconf" Resource by the Mediator, the Device establishes a  
310 TLS connection with the OCF Cloud using the URI that was provisioned, and the Device's  
311 manufacturer certificate and the trust anchor certificate(s) for OCF Cloud certificate validation, both  
312 of which were installed by the Device manufacturer. The combination of the Device's manufacturer  
313 certificate and OCF Cloud User's Access Token ensures the interactions between the OCF Cloud  
314 and OCF Devices are within the OCF Cloud User's domain.

315 To register with the OCF Cloud, the Device then sends an UPDATE operation to the Account  
316 Resource on the OCF Cloud which includes the Access Token that was provisioned in the  
317 "oic.r.coapcloudconf" Resource. Note that the OCF Cloud maintains a unique instance of the  
318 Account Resource for every Device.

319 If the UPDATE is successfully validated, then the OCF Cloud provides an UPDATE response that  
320 may provide updated values for the Access Token and details on the lifetime (expiration) of that  
321 Token. The OCF Cloud also includes the User ID to which the Device is associated. All values  
322 returned are stored securely on the Device. The returned Access Token is not written to the  
323 "oic.r.coapcloudconf" Resource.

324 The Device is now registered with the OCF Cloud.

#### 325 **5.4.5 Connection with the OCF Cloud**

326 See 8.1.4, see also ISO/IEC 30118-2:2018 clause 13.12

327 In order to enable passing data between the Device and the OCF Cloud, the Device sends an  
328 UPDATE request to the Session Resource; once validated, the OCF Cloud sends a response  
329 message that includes the remaining lifetime of the associated Access Token. The Device now has  
330 an active connection and can exchange data.

#### 331 **5.4.6 Publishing Links to the OCF Cloud RD**

332 See 8.2; see also ISO/IEC 30118-2:2018 clause 10.5, ISO/IEC 30118-1:2018 clause 11.3.6.

333 Once the TLS connection has been established to the OCF Cloud the Device exposes its Resources  
334 in the Resource Directory in the OCF Cloud so that they may be seen/accessed remotely.

#### 335 **5.4.7 Client to Server communication through the OCF Cloud**

336 See 8.3, 8.4; see also ISO/IEC 30118-2:2018 clause 10.5.

337 As for a Server, Clients follow this same process and register with the OCF Cloud.

338 The OCF Cloud allows communication between all of an OCF Cloud User's Devices based on the  
339 fact that they have the same User ID.

340 When the Client attempts CRUDN actions on the Links hosted by the OCF Cloud, the OCF Cloud  
341 forwards those requests to the Device. The Device responds to the OCF Cloud which then proxies  
342 the response to the C

343 lient (i.e. Client -> OCF Cloud -> Device -> OCF Cloud -> Client).



344 **5.4.8 Refreshing connection with the OCF Cloud**

345 See ISO/IEC 30118-2:2018 clause 13.13.

346 When (or before) the Access Token expires, the Device refreshes its token by sending an UPDATE  
347 request to the Token Refresh Resource.

348 **5.4.9 Closing connection with the OCF Cloud**

349 See ISO/IEC 30118-2:2018 clause 13.12.

350 To log out of the OCF Cloud the Device sends an UPDATE request to the Session Resource  
351 indicating a "login" status of "false". This does not delete or remove any of the Device Registration  
352 information. The Device may log back into the OCF Cloud at any point prior to expiration of the  
353 Access Token.

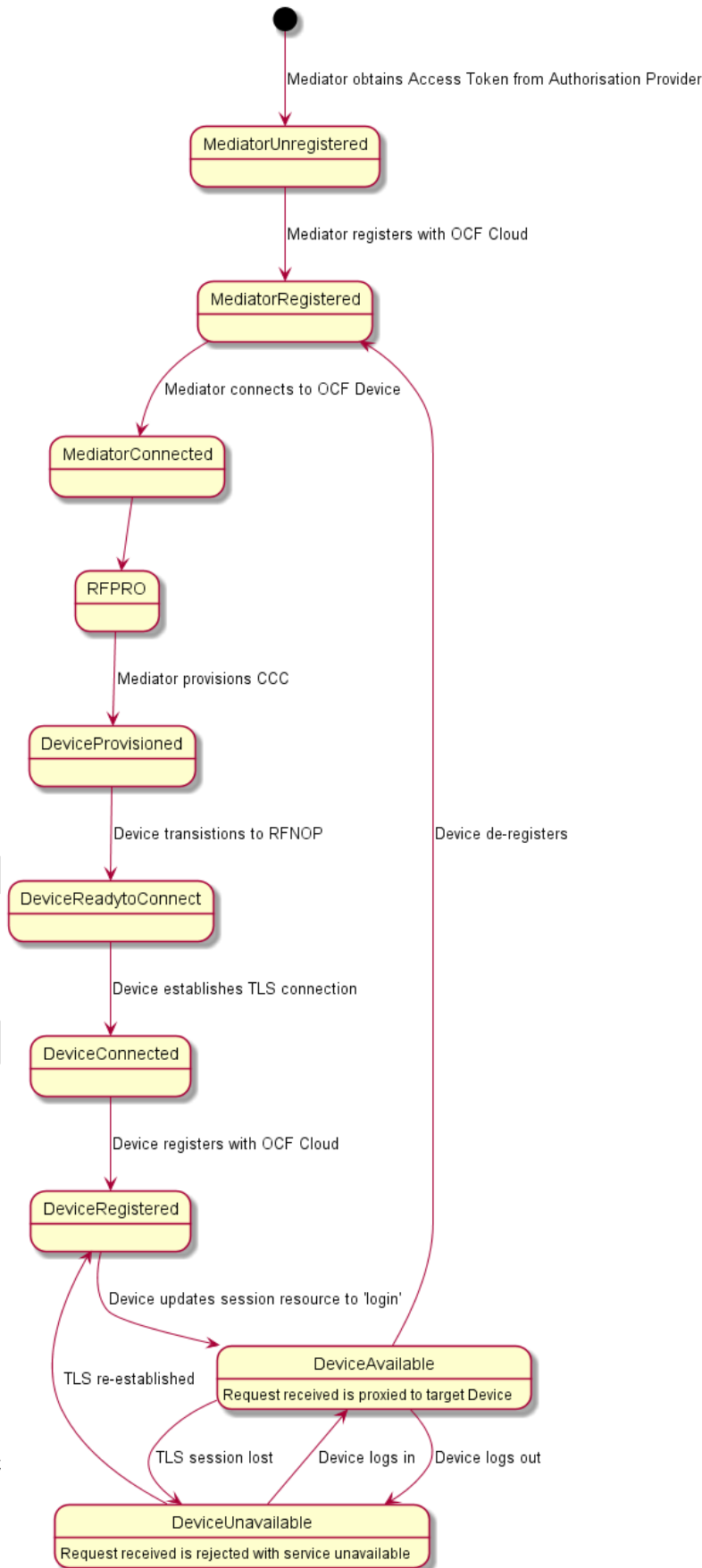
354 **5.4.10 Deregistering from the OCF Cloud**

355 See 8.5; see also ISO/IEC 30118-2:2018 clause 13.10.

356 To deregister with the OCF Cloud, the Device sends a DELETE request message to the Account  
357 Resource including its Access Token. The OCF Cloud sends a response message confirming that  
358 the Device has been deregistered.

359 To connect to the OCF Cloud again, the Device has to re-follow the flow starting with Mediator  
360 provisioning (see 5.4.3).

361 Figure 3 captures the state machine that is described by the informative operation flow provided in  
362 5.4.



**Figure 3 – Overall Operational State Machine**

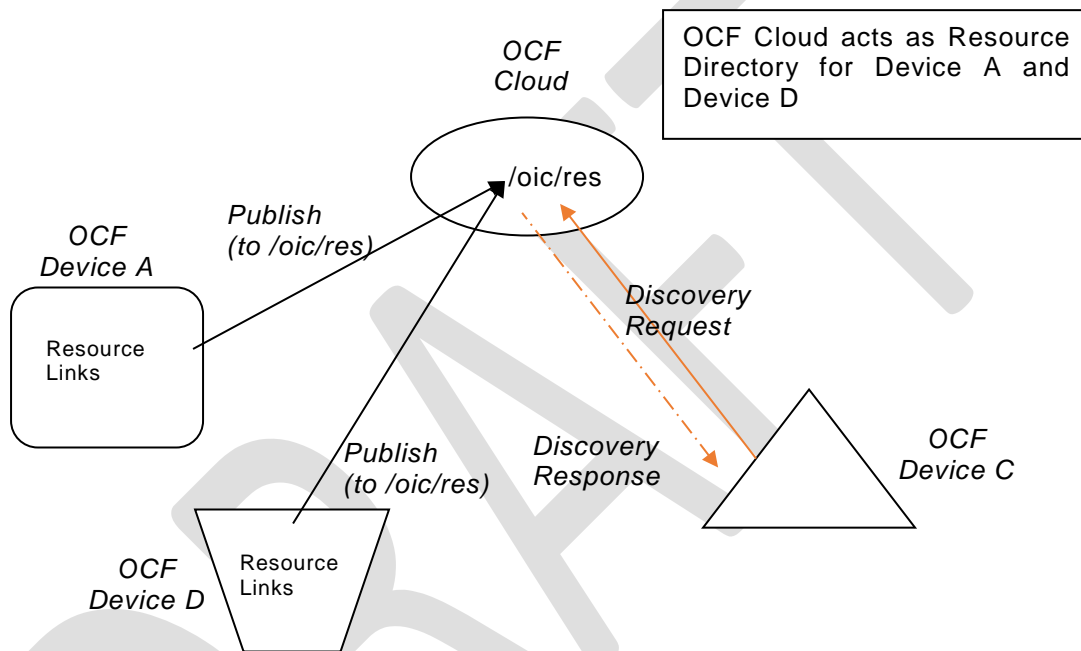
364

365 **6 Resource model**

366 **6.1 OCF Cloud Resource Directory**

367 **6.1.1 Indirect discovery for lookup of the Resources**

368 Indirect discovery is when a 3rd party, other than the discovering Device and the discovered Device,  
 369 assists with the discovery process. The 3rd party, called a Resource Directory (RD), only provides  
 370 information on Resources on behalf of another Device but does not host Resources on part of that  
 371 Device.



372

**Figure 4 – Indirect discovery of Resources by via an RD**

373

374 In Figure 4, the OCF Cloud acts as Resource Directory for Device A and Device D which are both  
 375 part of the same account. Device A and Device D publish their Resource information to the OCF  
 376 Cloud. Device C which is also part of the same account as Devices A and D, may query the OCF  
 377 Cloud to acquire the Resource information of Devices A and D.

378 Indirect discovery is useful for when Devices may not be on the same network and require  
 379 optimization for discovery or routing. Once Resources are discovered using indirect discovery, i.e.,  
 380 RD query, then the access to the Resource is done by a request sent to the endpoint exposed by  
 381 the RD for the Resource.

382 **6.1.2 Resource Directory Definition**

383 An OCF Cloud which acts as a Resource Directory (RD) will be involved in the following operations.

- 384 – *RD discovery* – the procedure by which publishing Devices discover an RD, in the case of the  
 385 OCF Cloud this is a direct result of Device registration with an OCF Cloud.

386 – *Resource publish* – the procedures with which Devices publish their Resource information, i.e.  
 387 Links.

388 – *Resource exposure* – the feature with which RDs expose the Links hosted by the 3<sup>rd</sup> party  
 389 Devices via their own "/oic/res".

390 An RD makes use of Resource Type "oic.wk.rd" defined in Table 2 and Table 3. An OCF Cloud that  
 391 supports the capability to host indirect discovery shall expose an instance of the "oic.wk.rd"  
 392 Resource Type in its "/oic/res" to announce that it serves as an RD. The use of the "oic.wk.rd"  
 393 Resource Type is restricted to OCF Clouds only, a proximal network Device shall not expose the  
 394 "oic.wk.rd" Resource Type.

395 The discoverable instance of "oic.wk.rd" shall allow only secure connections (e.g. OCF Endpoint  
 396 with a scheme of "coaps" or "coaps+tcp"). A publishing Device sends an UPDATE request to  
 397 "/oic/rd" with its Links in the payload to publish the Links in "/oic/res" of the RD. A publishing Device  
 398 is responsible for ensuring the RD has the correct published Links exposed via its "/oic/res".

399 **Table 2 – "oic.wk.rd" Resource Type definition**

Pre-defined URI	Resource Type Title	Resource Type ID ("rt" value)	OCF Interfaces	Description	Related Functional Interaction
"/oic/rd"	Resource Directory	"oic.wk.rd"	"oic.if.baseline"	The Discoverable Resource Type through which an RD 1) facilitates its discovery and provides the criteria to select an RD and 2) allows Devices to publish their Links in "/oic/res" of the RD.	Discovery

400

401

402

**Table 3 – "oic.wk.rd" Properties**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Selector	"sel"	"integer"	N/A	N/A	R	Yes	Provides the criteria for RD selection. An integer representing a value calculated by the RD. The value is in the range of 0 to 100. The lower the value, the more preferable the RD is.

403

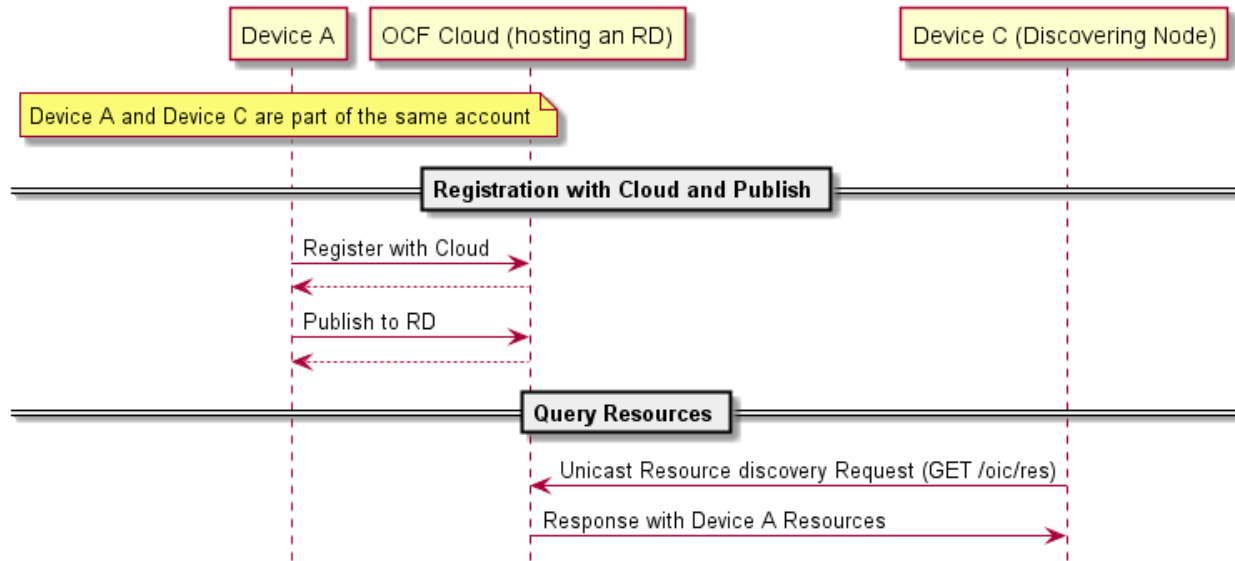
404 An RD may be queried at its "/oic/res" Resource to find Resources hosted on other Devices. A  
 405 publishing Device may publish all or a partial list of Resources they host to an RD. The RD then  
 406 responds to queries for Resource discovery on behalf of the publishing Device. Note that only  
 407 Devices that belong to the same account as the querying Device are visible in the exposed instance  
 408 of "/oic/res". For general Resource discovery, the RD behaves like any other Server in responding  
 409 to requests to "/oic/res".

410 **6.1.3 RD operational flows**

411 **6.1.3.1 Discovering an RD**

412 In Figure 5, a Device that wishes to publish its Resources first registers with the OCF Cloud that  
413 hosts the RD and then publishes the desired Resource information.

414



415

416

**Figure 5 – RD discovery and RD supported query of Resources support**

417 A Client that performs Resource discovery via an OCF Cloud RD does so via a unicast request to  
418 the RD; the Resource Directory defined in this document does not support the use of multicast  
419 queries to discover instances of an RD.

420 **6.1.3.2 Publish Resources**

421 **6.1.3.2.1 Overview**

422 After the selection process of an RD, a Device may push its Resource information to the selected  
423 RD, i.e., publish the Links in its "/oic/res" to the "/oic/res" of the RD.

424 The publishing Device may decide to publish all Resources or just a few of the Resources on the  
425 RD. At a minimum a publishing Device shall publish the mandatory Core Resources "/oic/d" and  
426 "/oic/p" as well as Resources that are defined as mandatory for the Device Type being published.  
427 The publishing Device should only publish Resources that are otherwise published to its own  
428 "/oic/res"; a publishing Device should not publish non-Discoverable Resources or Resources  
429 hosted by some other Device. A publishing Device shall respond to discovery requests on its  
430 "/oic/res" Resource unless all its Discoverable Resources have been published in an RD.

431 **6.1.3.2.2 Publish: Push Resource information**

432 Resource information may be published using an UPDATE request sent to "/oic/rd".

433 A Device which hosts a Resource may publish the Resource information, i.e. the Link targeting the  
434 Resource, to an RD by sending an UPDATE request with the Link in the payload. The published  
435 Link shall be exposed through the "/oic/res" of the RD.

436 When a Device first publishes a Link or Links, it shall send an UPDATE request to the "/oic/rd"  
437 Resource of the RD including the following key-value pairs in the payload:

- 438 – "di" –its value shall be the Device ID of the publishing Device, i.e. the "di" value of "/oic/d".
- 439 – "links" –its value shall be the array of Links to be published. Links may omit the "ins" Parameter  
440 in which case the RD will assign a value for each Link. The supplied "ins" Parameter by the  
441 Client is allowed to be overruled by the RD, e.g. an RD can ignore the supplied "ins" value.
- 442 – "ttl" –its value indicates how long (in seconds) the publishing Device requests the RD to keep  
443 this published Link.

444 Notice that the payload shall carry the appropriate Content-Format of "application/vnd.ocf+cbor".

```
445 {  
446   "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",  
447   "links": [  
448     {  
449       "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9"  
450       "href": "/myLightSwitch",  
451       "rt": ["oic.r.switch.binary"],  
452       "if": ["oic.if.a", "oic.if.baseline"],  
453       "p": {"bm": 3},  
454       "eps": [  
455         {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2},  
456         {"ep": "coaps://[fe80::b1d6]:1122"},  
457         {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}  
458       ]  
459     },  
460     {  
461       "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",  
462       "href": "/myLightBrightness",  
463       "rt": ["oic.r.brightness"],  
464       "if": ["oic.if.a", "oic.if.baseline"],  
465       "p": {"bm": 3},  
466       "eps": [  
467         {"ep": "coaps://[[2001:db8:a::123]:2222"}  
468       ]  
469     }  
470   ],  
471   "ttl": 600  
472 }
```

473 When an RD receives this initial UPDATE request, it determines whether to grant the request or  
474 not. Upon granting the request, the RD shall send back an UPDATE response to the publishing  
475 Device. The response shall include a payload with the same information as the original UPDATE  
476 request with the following possible differences:

- 477 – For each Link, an "ins" Parameter shall be included in the response. The RD shall assign a  
478 unique "ins" value identifying the Link among all the Links it advertises. If the publishing Device  
479 included an "ins" value in the UPDATE request, the RD may use it as long as it doesn't match  
480 any existing "ins" value in the published Links.
- 481 – The "ttl" Property Value shall be assigned by the RD and it shall be included in the response.  
482 The RD should use the value included in the UPDATE request but may assign a value that is  
483 lower if it is not able to honour the requested "ttl" value. After this time elapses, the RD shall  
484 remove the Links. To keep a Link alive the publishing Device may update the "ttl" using the  
485 UPDATE schema.

486 The RD shall add the new Links to its "/oic/res" and expose them to a valid discovery query, i.e.  
487 RETRIEVE request:

```
488 {  
489   "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",  
490   "links": [  
491     {  
492       "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",  
493       "href": "/myLightSwitch",  
494       "rt": ["oic.r.switch.binary"],  
495       "if": ["oic.if.a", "oic.if.baseline"],  
496       "p": {"bm": 3},  
497       "eps": [  
498         {"ep": "coaps://[fe80:b1d6]:1111", "pri": 2},  
499         {"ep": "coaps://[fe80:b1d6]:1122"},  
500         {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}  
501       ],  
502       "ins": 11235  
503     },  
504     {  
505       "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",  
506       "href": "/myLightBrightness",  
507       "rt": ["oic.r.brightness"],  
508       "if": ["oic.if.a", "oic.if.baseline"],  
509       "p": {"bm": 3},  
510       "eps": [  
511         {"ep": "coaps://[[2001:db8:a::123]:2222"}  
512       ],  
513       "ins": 112358  
514     }  
515   ].  
516   "ttl": 600  
517 }
```

### 518 6.1.3.3 Resource exposure

#### 519 6.1.3.3.1 "/oic/res" and retrieving of the Resources

520 The "/oic/res" based discovery process for an OCF Cloud does not support the use of multicast. A  
521 registered Client may discover Resources by sending a unicast RETRIEVE to "/oic/res". Only those  
522 Resources for Devices that are registered with the same account as the Client are returned in a  
523 response to the RETRIEVE.

524 Interaction with Resources discovered using the RD is done using the same mechanism and  
525 methods as with Resources discovered by retrieving the "/oic/res" Resource of the Device hosting  
526 the Resources (e.g., connect to the exposed endpoint and perform CRUDN operations on the  
527 Resource).

528 The "/oic/res" response to a requesting Client includes the Links with the "anchor" Parameter  
529 containing an OCF URI. The "/oic/res" response has a single array of Links. Each Link shall contain  
530 the "anchor" Parameter of the value OCF URI where the authority component of <deviceId>  
531 indicates the Device hosting the target Resource.

532 For example, an RD may return the following to a Client.

```
533 [  
534   {  
535     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",  
536     "href": "/oic/res",
```

```
537     "rel": "self",
538     "rt": ["oic.wk.res"],
539     "if": ["oic.if.ll", "oic.if.baseline"],
540     "p": {"bm": 3},
541     "eps": [
542       {"ep": "coap://[2001:db8:a::b1d4]:77777"},
543       {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
544     ],
545   },
546   {
547     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
548     "href": "/oic/d",
549     "rt": ["oic.wk.d", "oic.d.fan"],
550     "if": ["oic.if.r", "oic.if.baseline"],
551     "p": {"bm": 3},
552     "eps": [
553       {"ep": "coap://[2001:db8:a::b1d4]:77777"},
554       {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
555     ]
556   },
557   {
558     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
559     "href": "/oic/p",
560     "rt": ["oic.wk.p"],
561     "if": ["oic.if.r", "oic.if.baseline"],
562     "p": {"bm": 3},
563     "eps": [
564       {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
565     ]
566   },
567   {
568     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
569     "href": "/myFanIntrospection",
570     "rt": ["oic.wk.introspection"],
571     "if": ["oic.if.r", "oic.if.baseline"],
572     "p": {"bm": 3},
573     "eps": [
574       {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
575     ]
576   },
577   {
578     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
579     "href": "/oic/rd",
580     "rt": ["oic.wk.rd"],
581     "if": ["oic.if.baseline"],
582     "p": {"bm": 3},
583     "eps": [
584       {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
585     ]
586   },
587   {
588     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
589     "href": "/myFanSwitch",
590     "rt": ["oic.r.switch.binary"],
591     "if": ["oic.if.a", "oic.if.baseline"],
592     "p": {"bm": 3},
593     "eps": [
594       {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
595     ]
596   }
597 }
```



```
596     ,
597     {
598         "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
599         "href": "/oic/d",
600         "rt": ["oic.wk.d", "oic.d.light"],
601         "if": ["oic.if.r", "oic.if.baseline"],
602         "p": {"bm": 3},
603         "eps": [
604             {"ep": "coap://[2001:db8:b::c2e5]:66666"},
605             {"ep": "coaps://[2001:db8:b::c2e5]:22222"}
606         ]
607     },
608     {
609         "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
610         "href": "/oic/p",
611         "rt": ["oic.wk.p"],
612         "if": ["oic.if.r", "oic.if.baseline"],
613         "p": {"bm": 3},
614         "eps": [
615             {"ep": "coaps://[2001:db8:b::c2e5]:22222"}
616         ]
617     },
618     {
619         "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
620         "href": "/myLightSwitch",
621         "rt": ["oic.r.switch.binary"],
622         "if": ["oic.if.a", "oic.if.baseline"],
623         "p": {"bm": 3},
624         "eps": [
625             {"ep": "coaps://[2001:db8:b::c2e5]:22222"}
626         ]
627     },
628     {
629         "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
630         "href": "/myLightBrightness",
631         "rt": ["oic.r.brightness"],
632         "if": ["oic.if.a", "oic.if.baseline"],
633         "p": {"bm": 3},
634         "eps": [
635             {"ep": "coaps://[2001:db8:b::c2e5]:22222"}
636         ]
637     }
638 ]
```

639

## 640 **6.2 CoAPCloudConf Resource**

### 641 **6.2.1 Introduction**

642 The CoAPCloudConf resource exposes configuration information for connecting to an OCF Cloud.  
643 This is an optional discoverable Resource, which may additionally be included within the Easy  
644 Setup Collection ("oic.r.easyssetup") and so used during the Easy Setup process as defined in  
645 OCF Wi-Fi Easy Setup.

646 The CoAPCloudConf Resource shall expose only secure Endpoints (e.g. CoAPS); see the  
647 ISO/IEC 30118-1:2018, clause 10.

648 **6.2.2 Resource Definition**

649 The CoAPCloudConf Resource is as defined in Table 4.

 650 **Table 4 – CoAPCloudConf Resource**

Example URI	Resource Type Title	Resource Type ID ("rt" value)	Interfaces	Description	Related Functional Interaction
"/example/CoapCloudConfResURI"	CoAPCloudConf	"oic.r.coapcloudconf"	"oic.if.rw", "oic.if.baseline"	Configuration information for connecting to an OCF Cloud. The Resource properties exposed are listed in Table 5.	N/A

651

652

DRAFT

653 Table 5 defines the details for the "oic.r.coapcloudconf" Resource Type.

 654 **Table 5 – oic.r.coapcloudconf Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Auth Provider Name	apn	String	N/A	N/A	RW	No	The name of the Authorisation Provider through which access token was obtained.
OCF Cloud interface URL	cis	String	uri	N/A	RW	Yes	URL of OCF Cloud.
Access Token	at	String	The Access Token is a string of at least one character	N/A	W <sup>1</sup>	Yes (in an UPDATE only)	Access token which is returned by an Authorisation Provider or OCF Cloud.
OCF Cloud UUID	sid	uuid	N/A	N/A	RW	Yes	The identity of the OCF Cloud
Last Error Code during Cloud Provisioning	clec	integer	enum	N/A	R	No	0: No Error, 1: Error response from the OCF Cloud, 2: Failed to connect to the OCF Cloud, 3: Failed to refresh Access Token, 4~254: Reserved, 255: Unknown error

<sup>1</sup> The Access Token is not included in a RETRIEVE response payload. It can only be the target of an UPDATE.

655

656 If the "clec" Property is implemented by a Device it shall have an initial value of 0 ("No error").

 657 **6.2.3 Error Handling**

 658 The "clec" Property of the CoAPCloudConf Resource (i.e. "oic.r.coapcloudconf") is used to indicate  
 659 any error that occurred in the cloud configuration process while trying to connect to the OCF Cloud  
 660 (using the information populated by the Mediator in the CoAPCloudConf Resource). This is an  
 661 optional Property and if implemented, is set by the Device:

 662 – The Device shall set the "clec" Property to 1 if it receives an error response from the OCF Cloud  
 663 (e.g. error response from the Cloud).

 664 – The Device shall set the "clec" Property to 2 if there is a failure to connect to the OCF Cloud  
 665 (e.g. no reply, timeout, or timeout).

- 666 – The Device shall set the "clec" Property to 3 if it fails to refresh the Access Token (e.g. if it  
667 receives an error response during the token refresh procedure).

668 **7 Network and connectivity**

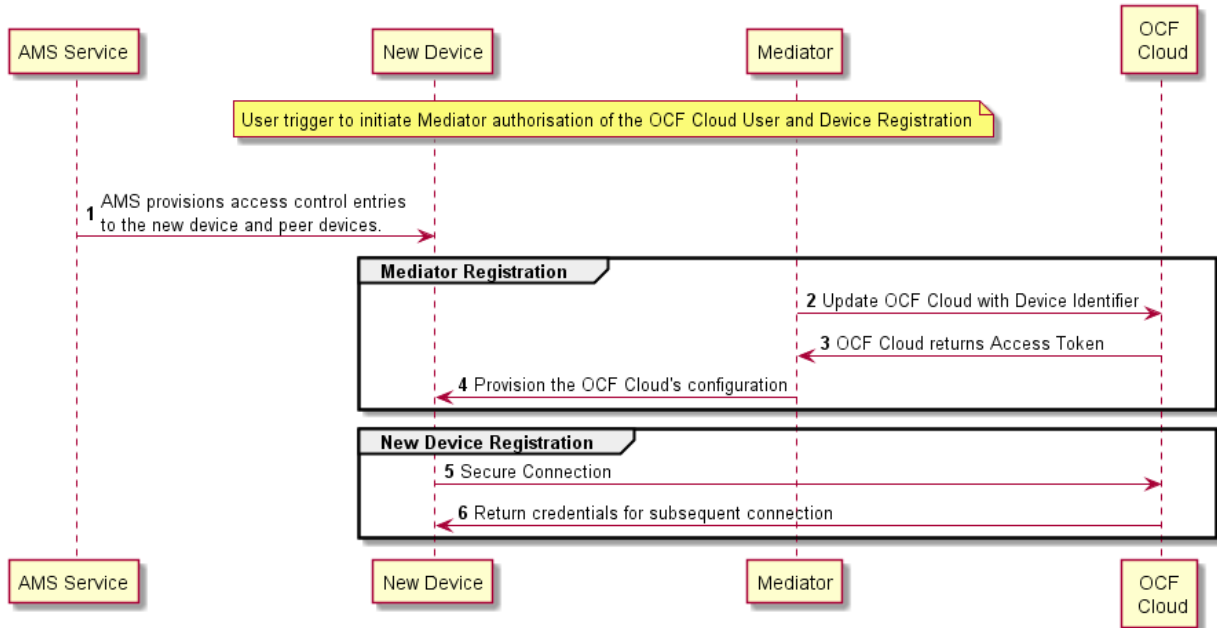
669 A TLS session exists between a Device and the OCF Cloud as specified in IETF RFC 8323; this is  
670 established following device configuration as detailed in 8.1.2.3.

DRAFT

671 **8 Functional interactions**

 672 **8.1 Onboarding, Provisioning, and Configuration**

 673 **8.1.1 Overview**

 674 Figure 6 provides an overview of the interaction between the different entities to get the Device  
 675 registered with the OCF Cloud. A summary of the flow is provided in Table 4.


676

677

**Figure 6 – Registration with OCF Cloud**

678

679

**Table 6 – Device to OCF Cloud Registration Flow**

Steps	Description
1	AMS provisions access control entries to the new device and peer devices.
2-3	Mediator obtains the OCF Cloud User's information and authorisation.
4	Mediator provisions the credentials for the Device to connect to the OCF Cloud
5-6	Device connects to the OCF Cloud using manufacturer certificate. The OCF Cloud returns credentials to the Device, used for subsequent connection to the OCF Cloud.

680

 681 **8.1.2 Use of Mediator**

 682 **8.1.2.1 Introduction**

 683 The Mediator is a specialised service that is used for provisioning the "oic.r.coapcloudconf"  
 684 Resource, and enabling connection of a headless Device to an OCF Cloud. The Mediator is  
 685 specified in OCF Wi-Fi Easy Setup.

686 The Mediator is implemented as part of the OBT (Onboarding Tool); and so could be part of any  
687 Device that itself hosts an OBT. A Device is authorized to communicate with an OCF Cloud if a  
688 trusted Mediator has provisioned the Device. The Device and Mediator connect over DTLS using  
689 credentials from "/oic/sec/cred".

690 As part of Device provisioning, the Mediator sets the following information in the  
691 "oic.r.coapcloudconf" Resource exposed by the Device:

- 692 – OCF Cloud Interface URL ("cis") Property
- 693 – OCF Cloud UUID ("sid") Property (to verify Cloud identity)
- 694 – Access Token ("at") Property that is validated by the OCF Cloud
- 695 – Optionally the Authorisation Provider name ("apn") Property through which the Access Token  
696 was obtained

697 If an error occurs during the process of registering and authenticating a Device with the OCF Cloud  
698 the Mediator may RETRIEVE the "clec" Property if implemented by the "oic.r.coapcloudconf"  
699 Resource on the Device to obtain a hint as to the cause of the error.

#### 700 **8.1.2.2 OCF Cloud User Authorisation of the Mediator**

701 The Mediator uses a user authorisation mechanism to enable the OCF Cloud to validate the OCF  
702 Cloud User's authorisation and obtain the OCF Cloud User's identity. The Authorisation Provider  
703 should be trusted by both the OCF Cloud User and the OCF Cloud. The Mediator may use OAUTH  
704 2.0 (see IETF RFC 6749) or another user authentication mechanism to obtain an Access Token as  
705 a form of authorisation from an OCF Cloud User via an Authorisation Provider. This authorisation  
706 achieves a variety of purposes. Firstly, the authorisation shows OCF Cloud User consent for  
707 Mediator to connect to the OCF Cloud. Secondly, the authorisation is used to obtain information to  
708 map the Devices to the same OCF Cloud User.

709 A user authorisation mechanism is used to achieve the following:

- 710 – Obtain an Access Token that is validated by the Cloud
- 711 – OCF Cloud User authorisation via an Authorisation Provider; this provides consent to connect  
712 to the OCF Cloud.

713 If a different Mediator is used by the same OCF Cloud User, a new Access Token may be obtained  
714 from an Authorisation Provider. Mediator Registration with the OCF Cloud

715 The Mediator connects to the OCF Cloud using a provisioned certificate on the Mediator to establish  
716 a TLS connection.

717 On its first connection, the Mediator starts the registration process with the OCF Cloud. The  
718 Mediator provides the OCF Cloud with the Mediator's Access Token received from the Authorisation  
719 Provider in 8.1.2.2 in order to register with the OCF Cloud.

720 The OCF Cloud then verifies the Access Token with the Authorisation Provider. If the Authorisation  
721 Provider validates the Access Token successfully, then it will return information about the OCF  
722 Cloud User to whom the Access Token belongs. The OCF Cloud generates a unique Access Token  
723 for the Mediator (which may be the original Access Token from the Mediator or a new Access Token)  
724 and a User ID (i.e. "uid" Property of "oic.r.account") if this is the first instance of registering a  
725 Mediator with this OCF Cloud User. The User ID acts as a unique identity for the OCF Cloud User.  
726 All instances of a Mediator for the same OCF Cloud User will be associated with the same User ID.  
727 This information is returned to the Mediator over TLS. The returned Access Token and User ID are

728 used by the OCF Cloud to identify the Mediator. This returned Access Token is used by the  
729 Mediator in subsequent interactions with the OCF Cloud.

730 All Devices registering with the OCF Cloud receive the same User ID from the OCF Cloud when  
731 registering with the same Mediator.

### 732 8.1.2.3 Device Provisioning by the Mediator

733 The Mediator obtains the OCF Cloud User's permission before the Mediator and OCF Cloud interact  
734 to preregister the Device with the OCF Cloud. This clause provides an informative description of  
735 the expected subsequent exchange between a Mediator and an OCF Cloud.

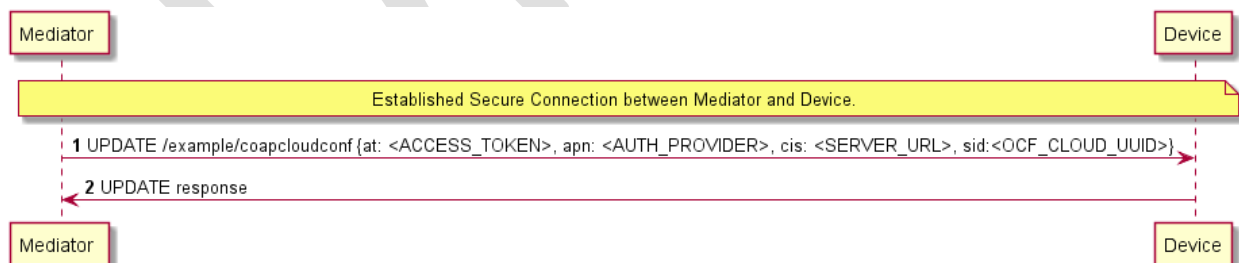
736 Once the OCF Cloud has associated the Mediator with a User ID, the Mediator can request the  
737 OCF Cloud to associate OCF Devices with the same User ID. To register the Device with the OCF  
738 Cloud, the Mediator first requests an Access Token for the Device from the OCF Cloud. The  
739 Mediator may provide the following information to the OCF Cloud to obtain an Access Token for  
740 the Device:

- 741 – Device ID (i.e. "di" Property Value of "/oic/d" of the Device)

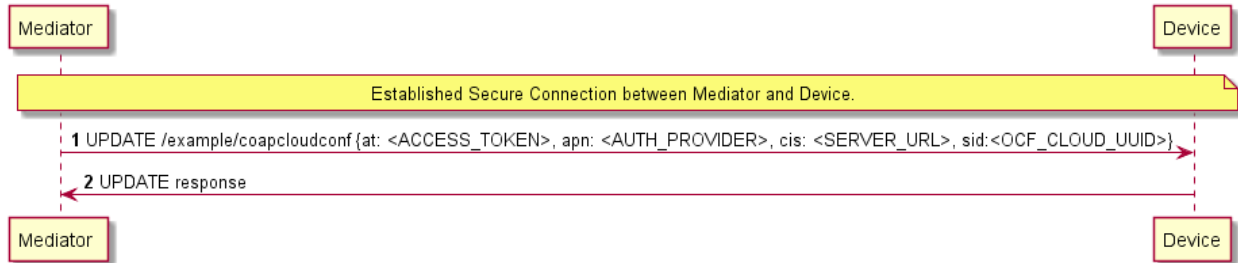
742 The OCF Cloud then returns a unique Access Token for the Device. The OCF Cloud maintains a  
743 map where Access Token and Mediator-provided Device ID are stored. At the time of Device  
744 Registration OCF Cloud validates the Access Token and associates the TLS session with  
745 corresponding Device ID. The OCF Cloud may also return an Authorisation Provider Name  
746 associated with the Access Token if the Access Token for the Device was created by an entity  
747 other than the OCF Cloud.

748 The Mediator provides this Access Token to the Device ("at" Property) via an UPDATE to the  
749 Device's "oic.r.coapcloudconf" Resource. The provisioned Access Token is to be treated by Device  
750 as an Access Token with "Bearer" token type as defined in IETF RFC 6750. The Mediator also  
751 provisions the OCF Cloud URI ("cis" Property), where the OCF Cloud URI can be either pre-  
752 configured or provided to the Mediator via OCF Cloud User input. The Mediator further provisions  
753 the OCF Cloud UUD ("sid" Property) to the identity of the OCF Cloud. If the OCF Cloud also  
754 returned an Authorisation Provider Name in association with the Access Token for the Device then  
755 this is also provisioned by the Mediator on the Device ("apn" Property of "oic.r.coapcloudconf").

756 See ISO/IEC 30118-2:2018 clause 7.5.2 for details on the population of ACE2 entries on the Device  
757 to allow CRUDN operations from the Mediator and OCF Cloud.



758  
759 Figure 7 describes the flow for provisioning of the Device by a Mediator. Table 7 provides additional  
760 context around the flow.



761  
762

**Figure 7 – Device Provisioning by the Mediator**

763

764

**Table 7 – Device Provisioning by the Mediator**

Steps	Description
1 - 2	Mediator updates the "oic.r.coapcloudconf" Resource on the Device with configuration information to enable the Device to connect to the OCF Cloud

765

766  
767

Please see ISO/IEC 30118-2:2018 clause 7.5.2 for further details on the mapping of Properties between the Device and OCF Cloud.

768

**8.1.3 Device Connection to the OCF Cloud**

769  
770  
771  
772

On conclusion of Device provisioning as defined in 8.1.2.3 and after transitioning to a state of RFNOP (if not already in RFNOP) the Device shall establish a TLS connection with the OCF Cloud as defined in the ISO/IEC 30118-2:2018 clause 10.5. Further see the ISO/IEC 30118-2:2018 clause 10.5.3 for additional security considerations.

773  
774  
775  
776  
777  
778

If authentication of the TLS session being established as defined in the ISO/IEC 30118-2:2018 fails, the "clec" Property of the "oic.r.coapcloudconf" Resource on the Device (if supported) shall be updated about the failed state. If authentication succeeds, the Device and OCF Cloud establish an encrypted link in accordance with the negotiated cipher suite. Further, if the TLS connection is lost due to a failure the "clec" Property of the "oic.r.coapcloudconf" Resource on the Device (if supported) should be updated about the failed state (value of "2").

779  
780  
781  
782

If the TLS connection is lost either via a failure or closed by the OCF Cloud then it may be re-established by following the procedures in the ISO/IEC 30118-2:2018 clause 10.5. A Device may automatically attempt to re-establish the TLS connection, alternatively a Device may require some user trigger to initiate the re-establishment of the TLS connection.

783

**8.1.4 Device Registration with the OCF Cloud**

784  
785  
786  
787

The OCF Cloud maintains a map of User IDs ("uid" Property of "oic.r.account"), Device IDs ("di" Property of "oic.r.account") and Access Tokens ("accesstoken" Property of "oic.r.account"; populated with the same value as the "at" Property obtained from "oic.r.coapcloudconf") to authenticate Devices connecting to the OCF Cloud.

788  
789  
790  
791  
792

After the TLS connection is established with the OCF Cloud, the Device shall register with the OCF Cloud by sending an UPDATE request to "/oic/sec/account" as defined in clause 13.10 of the ISO/IEC 30118-2:2018. The OCF Cloud consequently associates the TLS connection with the corresponding "uid" and "di" Properties populated in the "/oic/sec/account/" Resource. Any other Device registering with the OCF Cloud is assigned the same User ID by the OCF Cloud when



793 registering with any Mediator associated with that User ID. Device Registration permits a Client to  
794 access Resources on the OCF Cloud which are associated with the same User ID as the Client.

795 If the Property values in the UPDATE to "/oic/sec/account" do not match the equivalents provided  
796 to the Mediator by the OCF Cloud the OCF Cloud should close the TLS connection with the Device.  
797 Note that the OCF Cloud may also apply additional out-of-band measures, for example the OCF  
798 Cloud may send an email to the OCF Cloud User for additional verification to register the Device.

799 If the UPDATE operation is accepted by the OCF Cloud, the OCF Cloud responds as defined in  
800 clause 13.10 of the ISO/IEC 30118-2:2018.

801 The "accesstoken" Property that is returned in the UPDATE response may be valid for limited  
802 duration; in this instance the Device may use the "/oic/sec/tokenrefresh" Resource to renew the  
803 "accesstoken" before the Access Token expires at the time specified in the "expiresin" Property.

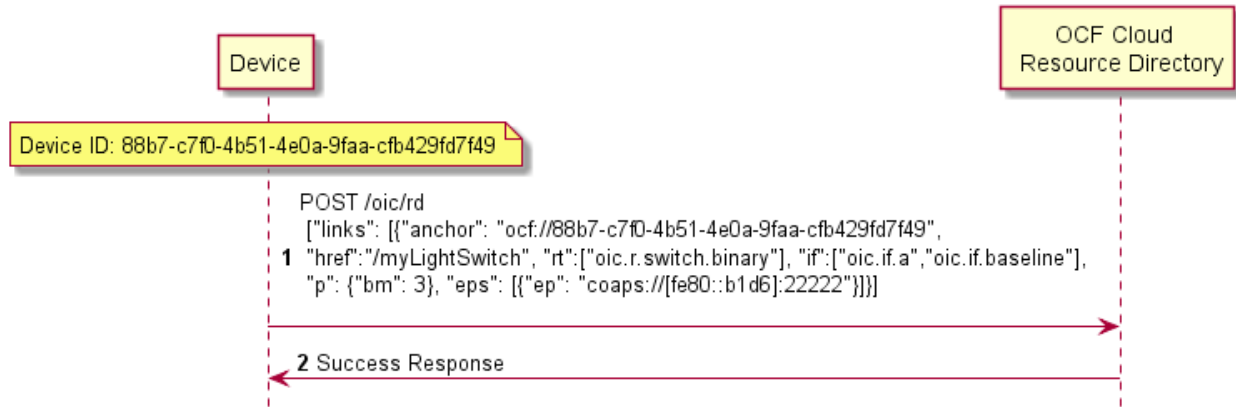
804 On completion of Device Registration the Device shall send an UPDATE to "/oic/sec/session" as  
805 defined in clause 13.11 of the ISO/IEC 30118-2:2018 to ensure that the established TLS session  
806 is maintained for subsequent interaction with the OCF Cloud Resource Directory as defined in  
807 clause 8.2.

## 808 **8.2 Resource Publication**

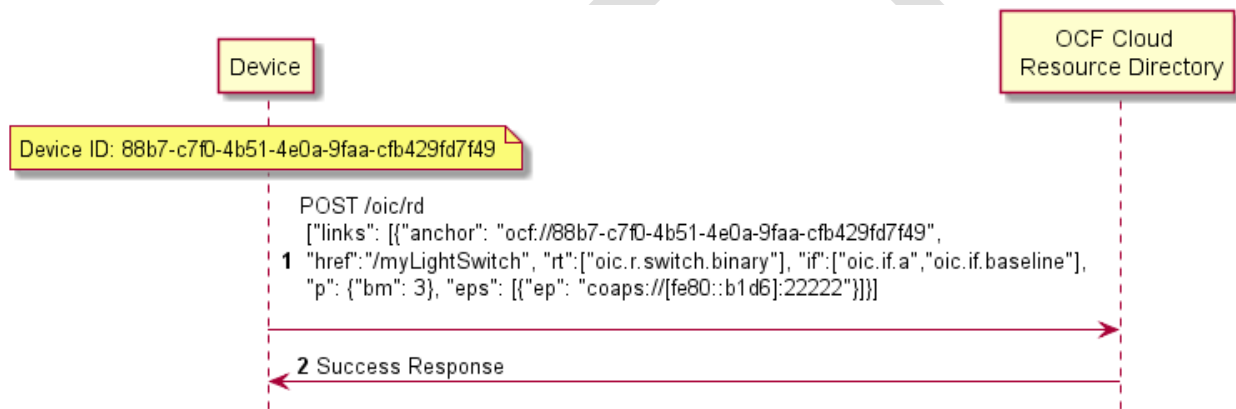
809 An OCF Cloud exposes a Resource Directory as defined in the ISO/IEC 30118-1:2018 clause  
810 11.3.6. After a Device is registered with an OCF Cloud, the Device should publish its Resources to  
811 the OCF Cloud's Resource Directory following the procedures defined in the ISO/IEC 30118-1:2018  
812 clause 11.3.6. The Device and OCF Cloud maintain a persistent TLS connection over which  
813 requests received by the OCF Cloud for the Device are routed.

814 The OCF Cloud maintains an internal association between the published Endpoint information from  
815 the Device and the Endpoint information that it (the OCF Cloud) exposes in the Links within the  
816 OCF Cloud's Resource Directory. The Endpoint exposed by the OCF Cloud for all Resources  
817 published to it is that of the OCF Cloud itself and not the publishing Device. These Endpoints use  
818 a scheme of "coaps+tcp". The Links within the OCF Cloud's Resource Directory are only identified  
819 per the OCF Cloud User Account (User ID). For example, the registered Links are only returned to  
820 Client under same User ID with a Server, and not returned to any other Client under a different  
821 User ID with the Server.

822 There is potential ambiguity where different instances of Devices from the same vendor (e.g.  
823 multiple lights) publish their Resources; this is because the local "href" Link Parameter that is  
824 provided to the RD is likely to be the same in each case. In order to avoid this ambiguity the  
825 Resource Directory shall prepend the "href" that is published with the Device ID for the publishing  
826 Device. Thus ensuring that all requests received by the OCF Cloud have a unique URI per  
827 published Resource.



828  
829 Figure 8 provides an example showing the provided Device ID from the Device; Figure 9 shows the  
830 pre-pending of the Device ID to the "href" Link Parameter in the Resource Directory itself.



831  
832 **Figure 8 – Resource publication to the OCF Cloud**

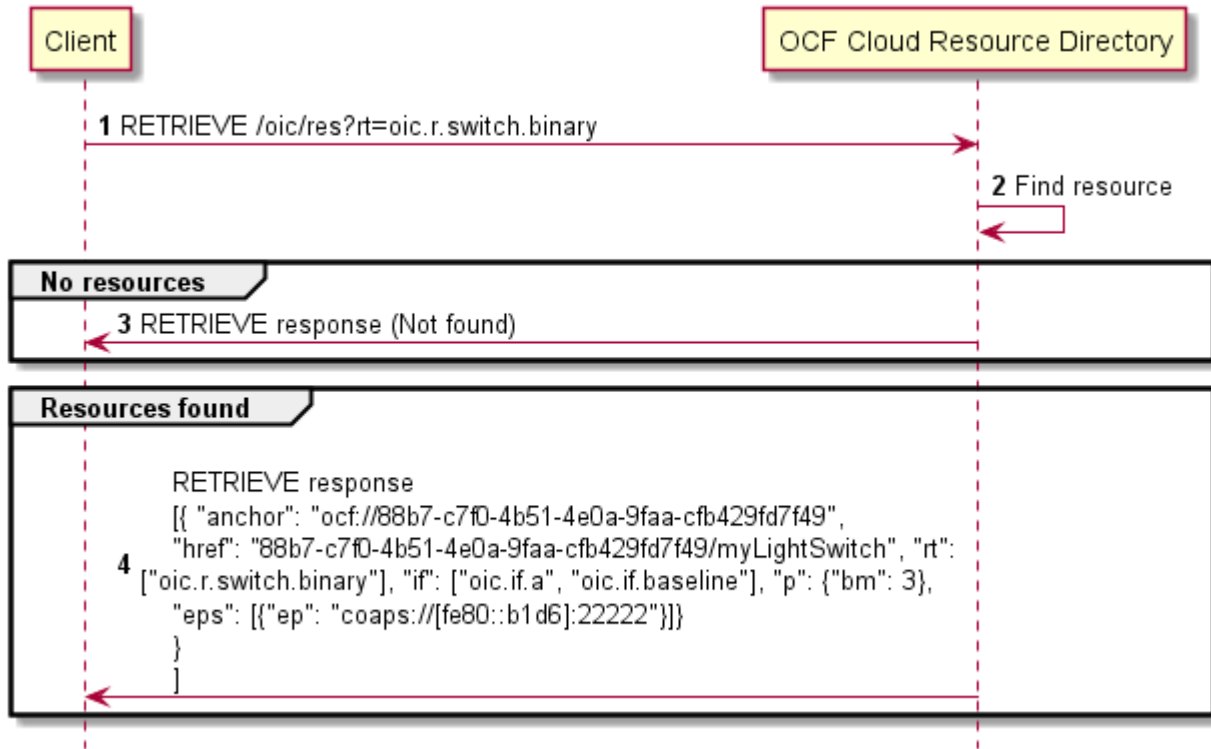
833 **8.3 Client Registration with the OCF Cloud**

834 A Device acting in the Client role follows the same procedures as a Device in the Server role  
835 registering with the OCF Cloud. This Client is associated with a User ID in the same manner in  
836 which a Server is associated with the same User ID

837 **8.4 Resource Discovery**

838 A remote Device may query "/oic/res" to discover Resources published to the OCF Cloud. The OCF  
839 Cloud's Resource Directory responds with Links for the Resources published to the OCF Cloud by  
840 Devices that are registered to the OCF Cloud for the User ID with which the remote Device is  
841 associated. The "eps" Link Parameter in the "/oic/res" response are for the OCF Cloud and not the  
842 publishing Device.

843 Figure 9 provides an illustrative flow for Resource Discovery, note the population of the 'href' for  
844 instance of "oic.r.switch.binary" including the Device ID of the target Device in accordance with 8.2:



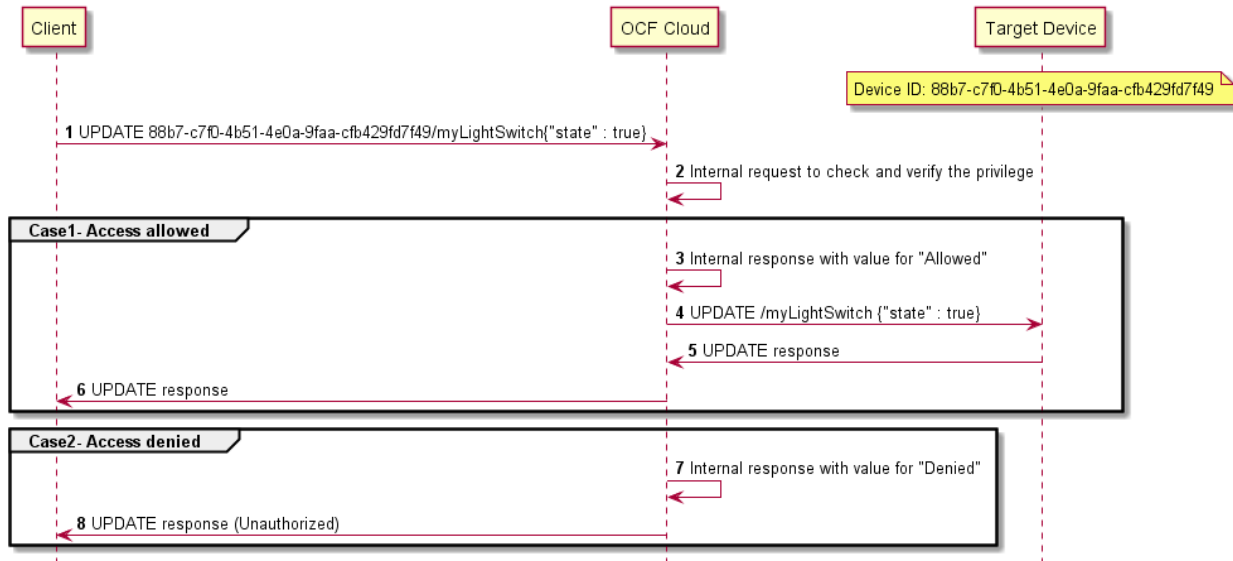
845

846

**Figure 9 – Resource discovery through OCF Cloud**

847 The OCF Cloud acts as a simple proxy, forwarding the messages to the publishing Devices. The  
 848 remote Device sends a RETRIEVE to the OCF Cloud to obtain the content of the Server's published  
 849 Resources, the OCF Cloud will route the message to the target Device after first removing the  
 850 Device ID that had been prepended to the 'href' Link Parameter by the Cloud RD. Similarly, other  
 851 CRUDN operations originated by a Client are routed to the Server via the OCF Cloud. The  
 852 publishing Device treats the forwarded request message as a request from the OCF Cloud. The  
 853 publishing Device authorises the request as specified in ISO/IEC 30118-2:2018, using the UUID of  
 854 the OCF Cloud configured in the "sid" Property of "oic.r.coapcloudconf". The publishing Device  
 855 sends a response message to the OCF Cloud, and the OCF Cloud forwards the response to the  
 856 Client which sent the corresponding request.

857 Figure 10 illustrates request routing via the OCF Cloud



858

859

**Figure 10 – Request routing through OCF Cloud**

860 If it is not possible for whatever reason for the OCF Cloud to route a Client request to the Server  
861 that OCF Cloud may reject the request with a final response (e.g. "Service Unavailable").

862 **8.5 Device Deregistration from the OCF Cloud**

863 To deregister from the OCF Cloud the Device first sends a DELETE operation to the  
864 "/oic/sec/account" Resource as defined in the ISO/IEC 30118-2:2018 clause 13.11.

865 Upon completion of deregistration of the Device the OCF Cloud deletes the links for the  
866 deregistered Device from the Resource Directory that is exposed by the OCF Cloud.

867 **9 Security**

868 OCF Cloud shall follow the security requirements captured in the ISO/IEC 30118-2:2018.

869

870  
 871  
 872

## Annex A (normative)

### Swagger2.0 definitions

873

#### A.1 List of Resource Type definitions

874

Table A.1 contains the list of defined resources in this document.

875

**Table A.1 – Alphabetized list of resources**

Friendly Name (informative)	Resource Type (rt)	Clause
Resource Directory	"oic.wk.rd"	A.2
CoAP Cloud Configuration	"oic.r.coapcloudconf"	A.3

876

#### A.2 Resource directory resource

877

##### A.2.1 Introduction

 878  
 879  
 880  
 881

Resource to be exposed by any Device that can act as a Resource Directory.  
 1) Provides selector criteria (e.g., integer) with GET request  
 2) Publish a Link in /oic/res with POST request

882

##### A.2.2 Well-known URI

883

/oic/rd

884

##### A.2.3 Resource type

885

The Resource Type is defined as: "oic.wk.rd".

886

##### A.2.4 OpenAPI 2.0 definition

 887  
 888  
 889  
 890  
 891  
 892  
 893  
 894  
 895  
 896  
 897  
 898  
 899  
 900  
 901  
 902  
 903  
 904  
 905  
 906  
 907  
 908  
 909  
 910  
 911  
 912  
 913

```
{
  "swagger": "2.0",
  "info": {
    "title": "Resource directory resource",
    "version": "2019-02-22",
    "license": {
      "name": "OCF Data Model License",
      "url":
        "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI
        CENSE.md",
      "x-copyright": "Copyright 2016-2019 Open Connectivity Foundation, Inc. All rights reserved."
    },
    "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
  },
  "schemes": ["http"],
  "consumes": ["application/json"],
  "produces": ["application/json"],
  "paths": {
    "/oic/rd" : {
      "get": {
        "description": "Resource to be exposed by any Device that can act as a Resource
        Directory.\n1) Provides selector criteria (e.g., integer) with GET request\n2) Publish a Link in
        /oic/res with POST request\n",
        "parameters": [
          {"$ref": "#/parameters/rdgetinterface"}
        ],
        "responses": {
```

```

914         "200": {
915             "description": "Respond with the selector criteria - either the set of attributes or
916 the bias factor\n",
917             "x-example": {
918                 "rt": ["oic.wk.rd"],
919                 "if": ["oic.if.baseline"],
920                 "sel": 50
921             },
922             "schema": { "$ref": "#/definitions/rdSelection" }
923         }
924     },
925 },
926 "post": {
927     "description": "Publish the Resource information for the first time in /oic/res. Updates to
928 existing entries are not allowed.\nAppropriates parts of the information, i.e., Links of the
929 published Resources will be discovered through /oic/res.\n1) When a Device first publishes a Link,
930 the request payload to RD may include the Links without an \"ins\" Parameter.\n2) Upon granting the
931 request, the RD assigns a unique instance value identifying the Link among all the Links it
932 advertises\n and sends back the instance value in the \"ins\" Parameter in the Link to the
933 publishing Device.\n",
934     "parameters": [
935         { "$ref": "#/parameters/rdpostinterface" },
936         {
937             "name": "body",
938             "in": "body",
939             "required": true,
940             "schema": { "$ref": "#/definitions/rdPublish" },
941             "x-example": {
942                 "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
943                 "links": [
944                     {
945                         "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
946                         "href": "/myLightSwitch",
947                         "rt": [ "oic.r.switch.binary" ],
948                         "if": [ "oic.if.a", "oic.if.baseline" ],
949                         "p": { "bm": 3 },
950                         "eps": [
951                             { "ep": "coaps://[2001:db8:a::b1d6]:1111", "pri": 2 },
952                             { "ep": "coaps://[2001:db8:a::b1d6]:1122" },
953                             { "ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3 }
954                         ]
955                     },
956                     {
957                         "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
958                         "href": "/myLightBrightness",
959                         "rt": [ "oic.r.brightness" ],
960                         "if": [ "oic.if.a", "oic.if.baseline" ],
961                         "p": { "bm": 3 },
962                         "eps": [
963                             { "ep": "coaps://[[2001:db8:a::123]:2222" }
964                         ]
965                     }
966                 ],
967                 "ttl": 600
968             }
969         }
970     ],
971     "responses": {
972         "200": {
973             "description": "Respond with the same schema as publish with the additional \"ins\"
974 Parameter in the Link.\n",
975             "x-example": {
976                 "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
977                 "links": [
978                     {
979                         "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
980                         "href": "/myLightSwitch",

```

```
981         "rt":      [ "oic.r.switch.binary" ],
982         "if":      [ "oic.if.a", "oic.if.baseline" ],
983         "p":       { "bm": 3 },
984         "eps": [
985             { "ep": "coaps://[2001:db8:a::b1d6]:1111", "pri": 2 },
986             { "ep": "coaps://[2001:db8:a::b1d6]:1122" },
987             { "ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3 }
988         ],
989         "ins":      11235
990     },
991     {
992         "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
993         "href":   "/myLightBrightness",
994         "rt":     [ "oic.r.brightness" ],
995         "if":     [ "oic.if.a", "oic.if.baseline" ],
996         "p":      { "bm": 3 },
997         "eps": [
998             { "ep": "coaps://[2001:db8:a::123]:2222" }
999         ],
1000        "ins":      112358
1001    }
1002 ],
1003 "ttl": 600
1004 },
1005 "schema": { "$ref": "#/definitions/rdPublish" }
1006 }
1007 }
1008 }
1009 }
1010 },
1011 "parameters": {
1012     "rdgetinterface" : {
1013         "in" : "query",
1014         "name" : "if",
1015         "type" : "string",
1016         "enum" : [ "oic.if.baseline" ]
1017     },
1018     "rdpostinterface" : {
1019         "in" : "query",
1020         "name" : "if",
1021         "type" : "string",
1022         "enum" : [ "oic.if.baseline" ]
1023     }
1024 },
1025 "definitions": {
1026     "rdSelection" : {
1027         "properties": {
1028             "rt" : {
1029                 "description": "Resource Type of the Resource",
1030                 "items": {
1031                     "enum": [ "oic.wk.rd" ],
1032                     "type": "string",
1033                     "maxLength": 64
1034                 },
1035                 "minItems": 1,
1036                 "uniqueItems": true,
1037                 "readOnly": true,
1038                 "type": "array"
1039             },
1040             "n" : {
1041                 "$ref":
1042 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1043 schema.json#/definitions/n"
1044             },
1045             "sel" : {
1046                 "description": "A bias factor calculated by the Resource Directory",
1047                 "maximum": 100,
```

```

1048         "minimum": 0,
1049         "readOnly": true,
1050         "type": "integer"
1051     },
1052     "id" : {
1053         "$ref":
1054         "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1055         schema.json#/definitions/id"
1056     },
1057     "if" : {
1058         "description": "The OCF Interfaces supported by this Resource",
1059         "items": {
1060             "enum": [
1061                 "oic.if.baseline"
1062             ],
1063             "type": "string",
1064             "maxLength": 64
1065         },
1066         "minItems": 1,
1067         "readOnly": true,
1068         "uniqueItems": true,
1069         "type": "array"
1070     }
1071 },
1072 "type" : "object",
1073 "required": ["sel"]
1074 },
1075 "rdPublish" : {
1076     "properties": {
1077         "di" : {
1078             "$ref":
1079             "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1080             schema.json#/definitions/di"
1081         },
1082         "ttl" : {
1083             "description": "Time to indicate a RD, i.e. how long to keep this published item.",
1084             "type": "integer"
1085         },
1086         "links" : {
1087             "description": "A set of simple or individual OCF Links.",
1088             "items": {
1089                 "properties": {
1090                     "anchor": {
1091                         "$ref":
1092                         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1093                         schema.json#/definitions/anchor"
1094                     },
1095                     "di": {
1096                         "$ref":
1097                         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1098                         schema.json#/definitions/di"
1099                     },
1100                     "eps": {
1101                         "$ref":
1102                         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1103                         schema.json#/definitions/eps"
1104                     },
1105                     "href": {
1106                         "$ref":
1107                         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1108                         schema.json#/definitions/href"
1109                     },
1110                     "if": {
1111                         "description": "The interface set supported by the published resource",
1112                         "items": {
1113                             "enum": [
1114                                 "oic.if.baseline",

```



```

1115         "oic.if.ll",
1116         "oic.if.b",
1117         "oic.if.rw",
1118         "oic.if.r",
1119         "oic.if.a",
1120         "oic.if.s"
1121     ],
1122     "type": "string",
1123     "maxLength": 64
1124 },
1125     "minItems": 1,
1126     "uniqueItems": true,
1127     "type": "array"
1128 },
1129     "ins": {
1130         "$ref":
1131         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1132         schema.json#/definitions/ins"
1133     },
1134     "p": {
1135         "$ref":
1136         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1137         schema.json#/definitions/p"
1138     },
1139     "rel": {
1140         "description": "The relation of the target URI referenced by the Link to the context
1141 URI",
1142         "oneOf": [
1143             {
1144                 "default": [
1145                     "hosts"
1146                 ],
1147                 "items": {
1148                     "maxLength": 64,
1149                     "type": "string"
1150                 },
1151                 "minItems": 1,
1152                 "type": "array"
1153             },
1154             {
1155                 "default": "hosts",
1156                 "maxLength": 64,
1157                 "type": "string"
1158             }
1159         ]
1160     },
1161     "rt": {
1162         "description": "Resource Type of the published Resource",
1163         "items": {
1164             "maxLength": 64,
1165             "type": "string"
1166         },
1167         "minItems": 1,
1168         "maxItems": 1,
1169         "uniqueItems": true,
1170         "type": "array"
1171     },
1172     "title": {
1173         "$ref":
1174         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1175         schema.json#/definitions/title"
1176     },
1177     "type": {
1178         "$ref":
1179         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1180         schema.json#/definitions/type"
1181     }

```

```

1182     },
1183     "required": [
1184       "href",
1185       "rt",
1186       "if"
1187     ],
1188     "type": "object"
1189   },
1190   "type": "array"
1191 }
1192 },
1193 "type" : "object",
1194 "required": ["di", "links", "ttl"]
1195 }
1196 }
1197 }
1198

```

### 1199 A.2.5 Property definition

1200 Table A-2 defines the Properties that are part of the "oic.wk.rd" Resource Type.

1201 **Table A-2 – The Property definitions of the Resource with type "rt" = "oic.wk.rd".**

Property name	Value type	Mandatory	Access mode	Description
rt	array: see schema	No	Read Only	Resource Type of the Resource.
n	multiple types: see schema	No	Read Write	
sel	integer	Yes	Read Only	A bias factor calculated by the Resource Directory.
id	multiple types: see schema	No	Read Write	
if	array: see schema	No	Read Only	The OCF Interfaces supported by this Resource.
di	multiple types: see schema	Yes	Read Write	
ttl	integer	Yes	Read Write	Time to indicate a RD, i.e. how long to keep this published item.
links	array: see schema	Yes	Read Write	A set of simple or individual OCF Links.

### 1202 A.2.6 CRUDN behaviour

1203 Table A-3 defines the CRUDN operations that are supported on the "oic.wk.rd" Resource Type.

1204 **Table A-3 – The CRUDN operations of the Resource with type "rt" = "oic.wk.rd".**

Create	Read	Update	Delete	Notify
	get	post		observe

## 1205 **A.3 CoAP Cloud Configuration Resource**

### 1206 **A.3.1 Introduction**

1207 The CoAPCloudConf Resource exposes configuration information for connecting to an OCF Cloud.  
1208

### 1209 **A.3.2 Example URI**

1210 /CoAPCloudConfResURI

### 1211 **A.3.3 Resource type**

1212 The Resource Type is defined as: "oic.r.coapcloudconf".

### 1213 **A.3.4 OpenAPI 2.0 definition**

```
1214 {  
1215   "swagger": "2.0",  
1216   "info": {  
1217     "title": "CoAP Cloud Configuration Resource",  
1218     "version": "20190327",  
1219     "license": {  
1220       "name": "OCF Data Model License",  
1221       "url":  
1222         "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI  
1223         CENSE.md",  
1224       "x-copyright": "Copyright 2018-2019 Open Connectivity Foundation, Inc. All rights reserved."  
1225     },  
1226     "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"  
1227   },  
1228   "schemes": ["http"],  
1229   "consumes": ["application/json"],  
1230   "produces": ["application/json"],  
1231   "paths": {  
1232     "/CoAPCloudConfResURI?if=oic.if.rw" : {  
1233       "get": {  
1234         "description": "The CoAPCloudConf Resource exposes configuration information for connecting  
1235         to an OCF Cloud.\n",  
1236         "parameters": [  
1237           {"$ref": "#/parameters/interface-all"}  
1238         ],  
1239         "responses": {  
1240           "200": {  
1241             "description": "",  
1242             "x-example":  
1243               {  
1244                 "rt": ["oic.r.coapcloudconf"],  
1245                 "apn": "github",  
1246                 "cis": "coaps+tcp://example.com:443",  
1247                 "sid": "987e6543-a21f-10d1-a112-421345746237",  
1248                 "clec": 0  
1249               },  
1250             "schema": { "$ref": "#/definitions/CoAPCloudConf" }  
1251           }  
1252         }  
1253       },  
1254       "post": {  
1255         "description": "Update properties of the CoAPCloudConf Resource.\n",  
1256         "parameters": [  
1257           {"$ref": "#/parameters/interface-all"},  
1258           {  
1259             "name": "body",  
1260             "in": "body",  
1261             "required": true,  
1262             "schema": { "$ref": "#/definitions/CoAPCloudConfUpdate" },  

```

```

1263         "x-example":
1264         {
1265             "at": "0f3d9f7fe5491d54077d",
1266             "apn": "github",
1267             "cis": "coaps+tcp://example.com:443",
1268             "sid" : "987e6543-a21f-10d1-a112-421345746237"
1269         }
1270     },
1271 ],
1272 "responses": {
1273     "200": {
1274         "description" : "",
1275         "x-example":
1276         {
1277             "apn": "github",
1278             "cis": "coaps+tcp://example.com:443",
1279             "sid" : "987e6543-a21f-10d1-a112-421345746237",
1280             "clec": 0
1281         },
1282         "schema": { "$ref": "#/definitions/CoAPCloudConf" }
1283     }
1284 }
1285 },
1286 ],
1287 "/CoAPCloudConfResURI?if=oic.if.baseline" : {
1288     "get": {
1289         "description": "The CoAPCloudConf Resource exposes configuration information for connecting
1290 to an OCF Cloud.\n",
1291         "parameters": [
1292             { "$ref": "#/parameters/interface-all" }
1293         ],
1294         "responses": {
1295             "200": {
1296                 "description" : "",
1297                 "x-example":
1298                 {
1299                     "rt": ["oic.r.coapcloudconf"],
1300                     "if" : ["oic.if.rw","oic.if.baseline"],
1301                     "apn": "github",
1302                     "cis": "coaps+tcp://example.com:443",
1303                     "sid" : "987e6543-a21f-10d1-a112-421345746237",
1304                     "clec": 0
1305                 },
1306                 "schema": { "$ref": "#/definitions/CoAPCloudConf" }
1307             }
1308         }
1309     },
1310     "post": {
1311         "description": "Update Properties of the CoAPCloudConf Resource.\n",
1312         "parameters": [
1313             { "$ref": "#/parameters/interface-all" },
1314             {
1315                 "name": "body",
1316                 "in": "body",
1317                 "required": true,
1318                 "schema": { "$ref": "#/definitions/CoAPCloudConfUpdate" },
1319                 "x-example":
1320                 {
1321                     "at": "0f3d9f7fe5491d54077d",
1322                     "apn": "github",
1323                     "cis": "coaps+tcp://example.com:443",
1324                     "sid" : "987e6543-a21f-10d1-a112-421345746237"
1325                 }
1326             }
1327         ],
1328         "responses": {
1329             "200": {

```

```

1330         "description" : "",
1331         "x-example":
1332         {
1333             "apn": "github",
1334             "cis": "coaps+tcp://example.com:443",
1335             "sid" : "987e6543-a21f-10d1-a112-421345746237",
1336             "clec": 0
1337         },
1338         "schema": { "$ref": "#/definitions/CoAPCloudConf" }
1339     }
1340 }
1341 }
1342 }
1343 },
1344 "parameters": {
1345     "interface-all" : {
1346         "in" : "query",
1347         "name" : "if",
1348         "type" : "string",
1349         "enum" : ["oic.if.rw","oic.if.baseline"]
1350     }
1351 },
1352 "definitions": {
1353     "CoAPCloudConf" : {
1354         "properties": {
1355             "rt" : {
1356                 "description": "Resource Type of the Resource",
1357                 "items": {
1358                     "enum": ["oic.r.coapcloudconf"],
1359                     "type": "string",
1360                     "maxLength": 64
1361                 },
1362                 "minItems": 1,
1363                 "uniqueItems": true,
1364                 "readOnly": true,
1365                 "type": "array"
1366             },
1367             "n" : {
1368                 "$ref":
1369 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1370 schema.json#/definitions/n"
1371             },
1372             "cis" : {
1373                 "description": "URL of OCF Cloud",
1374                 "format": "uri",
1375                 "type": "string"
1376             },
1377             "apn" : {
1378                 "description": "The Authorisation Provider through which an Access Token was obtained.",
1379                 "type": "string"
1380             },
1381             "sid" : {
1382                 "$ref": "http://openconnectivityfoundation.github.io/core/schemas/oic.types-
1383 schema.json#/definitions/uuid"
1384             },
1385             "clec" : {
1386                 "description": "Last Error Code during Cloud Provisioning (0: No Error, 1: Error response
1387 from the OCF Cloud, 2: Failed to connect to the OCF Cloud, 3: Failed to refresh Access Token, 4~254:
1388 Reserved, 255: Unknown error)",
1389                 "enum": [
1390                     0,
1391                     1,
1392                     2,
1393                     3,
1394                     255
1395                 ],
1396                 "readOnly": true

```

```

1397     },
1398     "id" : {
1399       "$ref":
1400       "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1401       schema.json#/definitions/id"
1402     },
1403     "if" : {
1404       "description": "The OCF Interfaces supported by this Resource",
1405       "items": {
1406         "enum": [
1407           "oic.if.rw",
1408           "oic.if.baseline"
1409         ],
1410         "type": "string",
1411         "maxLength": 64
1412       },
1413       "minItems": 2,
1414       "uniqueItems": true,
1415       "readOnly": true,
1416       "type": "array"
1417     }
1418   },
1419   "type" : "object",
1420   "required":["cis", "sid"]
1421 },
1422 "CoAPCloudConfUpdate" : {
1423   "properties": {
1424     "cis" : {
1425       "description": "URL of OCF Cloud",
1426       "format": "uri",
1427       "type": "string"
1428     },
1429     "apn" : {
1430       "description": "The Authorisation Provider through which an Access Token was obtained.",
1431       "type": "string"
1432     },
1433     "at" : {
1434       "description": "Access Token which is returned by an Authorisation Provider or OCF
1435       Cloud.",
1436       "type": "string"
1437     },
1438     "sid" : {
1439       "$ref": "http://openconnectivityfoundation.github.io/core/schemas/oic.types-
1440       schema.json#/definitions/uuid"
1441     }
1442   },
1443   "type" : "object",
1444   "required":["cis", "at", "sid"]
1445 }
1446 }
1447 }
1448

```

### 1449 A.3.5 Property definition

1450 Table A.4 defines the Properties that are part of the "oic.r.coapcloudconf" Resource Type.

1451 **Table A.4 – The Property definitions of the Resource with type "rt" = "oic.r.coapcloudconf".**

Property name	Value type	Mandatory	Access mode	Description
sid	multiple types: see schema	Yes	Read Write	
rt	array: see schema	No	Read Only	Resource Type of the Resource.

id	multiple types: see schema	No	Read Write	
n	multiple types: see schema	No	Read Write	
cis	string	Yes	Read Write	URL of OCF Cloud.
apn	string	No	Read Write	The Authorisation Provider through which an Access Token was obtained.
if	array: see schema	No	Read Only	The OCF Interfaces supported by this Resource.
clec	multiple types: see schema	No	Read Only	Last Error Code during Cloud Provisioning (0: No Error, 1: Error response from the OCF Cloud, 2: Failed to connect to the OCF Cloud, 3: Failed to refresh Access Token, 4~254: Reserved, 255: Unknown error).
sid	multiple types: see schema	Yes	Read Write	
at	string	Yes	Read Write	Access Token which is returned by an Authorisation Provider or OCF Cloud.
apn	string	No	Read Write	The Authorisation Provider through which an Access Token was obtained.
cis	string	Yes	Read Write	URL of OCF Cloud.

 1452 **A.3.6 CRUDN behaviour**

 1453 Table A.5 defines the CRUDN operations that are supported on the "oic.r.coapcloudconf" Resource  
 1454 Type.

 1455 **Table A.5 – The CRUDN operations of the Resource with type "rt" = "oic.r.coapcloudconf".**

Create	Read	Update	Delete	Notify
	get	post		observe

1456