

OCF Device to Cloud Services Specification

VERSION 2.2.3 | April 2021



OPEN CONNECTIVITY
FOUNDATION™

CONTACT admin@openconnectivity.org
Copyright Open Connectivity Foundation, Inc. © 2021.
All Rights Reserved.

Legal Disclaimer

2
3

4 NOTHING CONTAINED IN THIS DOCUMENT SHALL BE DEEMED AS GRANTING YOU ANY KIND
5 OF LICENSE IN ITS CONTENT, EITHER EXPRESSLY OR IMPLIEDLY, OR TO ANY
6 INTELLECTUAL PROPERTY OWNED OR CONTROLLED BY ANY OF THE AUTHORS OR
7 DEVELOPERS OF THIS DOCUMENT. THE INFORMATION CONTAINED HEREIN IS PROVIDED
8 ON AN "AS IS" BASIS, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW,
9 THE AUTHORS AND DEVELOPERS OF THIS SPECIFICATION HEREBY DISCLAIM ALL OTHER
10 WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, STATUTORY OR AT
11 COMMON LAW, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF
12 MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OPEN CONNECTIVITY
13 FOUNDATION, INC. FURTHER DISCLAIMS ANY AND ALL WARRANTIES OF NON-
14 INFRINGEMENT, ACCURACY OR LACK OF VIRUSES.

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

17 Copyright © 2018-2021 Open Connectivity Foundation, Inc. All rights reserved.

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

19

CONTENTS

20
21
22
23
24
25
26
27
28
29
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

Introduction	vi
1 Scope.....	1
2 Normative references	1
3 Terms, definitions, and abbreviated terms	3
3.1 Terms and definitions.....	3
3.2 Symbols and abbreviated terms.....	3
4 Document conventions and organization	3
4.1 Conventions.....	3
4.2 Notation.....	3
5 Overview	5
5.1 Introduction.....	5
5.2 OCF Cloud Architecture Alignment with ISO IEC 17789	5
5.3 Architecture	6
5.4 Interaction Flow	6
5.5 Cloud Operational Flow	7
5.5.1 Introduction.....	7
5.5.2 Pre-requisites and OCF Cloud User Account Creation.....	8
5.5.3 Mediator registration with the OCF Cloud	8
5.5.4 Device provisioning by the Mediator	8
5.5.5 Device Registration with the OCF Cloud.....	8
5.5.6 Connection with the OCF Cloud.....	9
5.5.7 Publishing Links to the OCF Cloud RD	9
5.5.8 Client to Server communication through the OCF Cloud.....	9
5.5.9 Refreshing connection with the OCF Cloud.....	9
5.5.10 Closing connection with the OCF Cloud.....	9
5.5.11 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 Resources.....	12
6.1.2 Resource Directory Definition	12
6.1.3 RD operational flows	13
6.2 CoAPCloudConf Resource	18
6.2.1 Introduction.....	18
6.2.2 Resource Definition.....	18
6.2.3 Cloud status governing state machine	19
6.2.4 Error Handling	22
7 Network and connectivity	22
8 Functional interactions.....	23
8.1 Onboarding, Provisioning, and Configuration	23
8.1.1 Overview	23
8.1.2 Use of Mediator	23

63	8.1.3	Device Connection to the OCF Cloud	26
64	8.1.4	Device Registration with the OCF Cloud	26
65	8.2	Resource Publication	26
66	8.3	Client Registration with the OCF Cloud.....	27
67	8.4	Resource Discovery	27
68	8.5	Device Deregistration from the OCF Cloud.....	29
69	8.6	Device Management.....	29
70	8.6.1	Behaviours on Device maintenance state changes.....	29
71	9	Security.....	30
72	Annex A (normative)	Swagger2.0 definitions.....	31
73	A.1	List of Resource Type definitions.....	31
74	A.2	Resource directory resource.....	31
75	A.2.1	Introduction.....	31
76	A.2.2	Well-known URI	31
77	A.2.3	Resource type.....	31
78	A.2.4	OpenAPI 2.0 definition	31
79	A.2.5	Property definition.....	36
80	A.2.6	CRUDN behaviour.....	36
81	A.3	CoAP Cloud Configuration Resource	36
82	A.3.1	Introduction.....	36
83	A.3.2	Example URI.....	36
84	A.3.3	Resource type.....	36
85	A.3.4	OpenAPI 2.0 definition	36
86	A.3.5	Property definition.....	40
87	A.3.6	CRUDN behaviour.....	41
88			
89			

90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105

Figures

Figure 1 – OCF Cloud Architecture.....	6
Figure 2 – OCF Cloud interaction model.....	7
Figure 3 – Overall Operational State Machine.....	12
Figure 4 – Indirect discovery of Resources by via an RD.....	12
Figure 5 – RD discovery and RD supported query of Resources support	14
Figure 6 – Device registration status state machine	21
Figure 7 – Registration with OCF Cloud.....	23
Figure 8 – Device Provisioning by the Mediator	25
Figure 9 – Resource publication to the OCF Cloud	27
Figure 10 – Resource discovery through OCF Cloud.....	28
Figure 11 – Request routing through OCF Cloud	29

Tables

106	
107	
108	Table 1 – OCF Cloud Interaction Flow7
109	Table 2 – "oic.wk.rd" Resource Type definition 13
110	Table 3 – "oic.wk.rd" Properties 13
111	Table 4 – CoAPCloudConf Resource 18
112	Table 5 – oic.r.coapcloudconf Resource Type definition..... 19
113	Table 6 – Device registration states20
114	Table 7 – Device to OCF Cloud Registration Flow23
115	Table 8 – Device Provisioning by the Mediator25
116	Table 9 – Actions on Device state change29
117	Table 10 – Default values for CoAPCloudConf Resource29
118	Table A.1 – Alphabetized list of resources 31
119	Table A-2 – The Property definitions of the Resource with type "rt" = "oic.wk.rd".36
120	Table A-3 – The CRUDN operations of the Resource with type "rt" = "oic.wk.rd".....36
121	Table A.4 – The Property definitions of the Resource with type "rt" = "oic.r.coapcloudconf". . 40
122	Table A.5 – The CRUDN operations of the Resource with type "rt" = "oic.r.coapcloudconf" ...41
123	

Introduction

125 This document, and all the other parts associated with this document, were developed in response
126 to worldwide demand for smart home focused Internet of Things (IoT) devices, such as appliances,
127 door locks, security cameras, sensors, and actuators; these to be modelled and securely controlled,
128 locally and remotely, over an IP network.

129 While some inter-device communication existed, no universal language had been developed for
130 the IoT. Device makers instead had to choose between disparate frameworks, limiting their market
131 share, or developing across multiple ecosystems, increasing their costs. The burden then falls on
132 end users to determine whether the products they want are compatible with the ecosystem they
133 bought into, or find ways to integrate their devices into their network, and try to solve interoperability
134 issues on their own.

135 In addition to the smart home, IoT deployments in commercial environments are hampered by a
136 lack of security. This issue can be avoided by having a secure IoT communication framework, which
137 this standard solves.

138 The goal of these documents is then to connect the next 25 billion devices for the IoT, providing
139 secure and reliable device discovery and connectivity across multiple OSs and platforms. There
140 are multiple proposals and forums driving different approaches, but no single solution addresses
141 the majority of key requirements. This document and the associated parts enable industry
142 consolidation around a common, secure, interoperable approach.

143 **1 Scope**

144 This document defines functional extensions to the capabilities defined in ISO/IEC 30118-1 to meet
145 the requirements of the OCF Cloud. This document specifies new Resource Types to enable the
146 functionality and any extensions to the existing capabilities defined in ISO/IEC 30118-1.

147 **2 Normative references**

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

152 ISO/IEC 30118-1 *Information technology -- Open Connectivity Foundation (OCF) Specification --*
153 *Part 1: Core specification*

154 <https://www.iso.org/standard/53238.html>

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

156 ISO/IEC 30118-2 *Information technology -- Open Connectivity Foundation (OCF) Specification --*
157 *Part 2: Security specification*

158 <https://www.iso.org/standard/74239.html>

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

160 ISO/IEC 17788 *Information technology – Cloud computing – Overview and vocabulary*

161 <https://www.iso.org/standard/60544.html>

162 ISO/IEC 17789 *Information technology – Cloud computing – Reference architecture*

163 <https://www.iso.org/standard/60545.html>

164 OCF Core Optional Framework, *Open Connectivity Foundation Core – Optional Specification,*
165 *Version 2.2.0*

166 Available at: https://openconnectivity.org/specs/OCF_Core_Optional_Specification_v2.2.0.pdf

167 Latest version available at:

168 https://openconnectivity.org/specs/OCF_Core_Optional_Specification.pdf

169 OCF Wi-Fi Easy Setup, *Open Connectivity Foundation Wi-Fi Easy Setup, Version 2.2.0*

170 Available at: https://openconnectivity.org/specs/OCF_Wi-Fi_Easy_Setup_Specification_v2.2.0.pdf

171 Latest version available at:

172 https://openconnectivity.org/specs/OCF_Wi-Fi_Easy_Setup_Specification.pdf

173 OCF Cloud Security, *Open Connectivity Foundation Cloud Security, Version 2.2.0*

174 Available at: https://openconnectivity.org/specs/OCF_Cloud_Security_Specification_v2.2.0.pdf

175 Latest version available at:

176 https://openconnectivity.org/specs/OCF_Cloud_Security_Specification.pdf

177 OCF Cloud API for Cloud Services, *Open Connectivity Foundation Cloud API for Cloud Services,*
178 *Version 2.2.0*

179 Available at:

180 https://openconnectivity.org/specs/OCF_Cloud_API_For_Cloud_Services_Specification_v2.2.0.pdf

181 f

182 Latest version available at:

183 https://openconnectivity.org/specs/OCF_Cloud_API_For_Cloud_Services_Specification.pdf

184 IETF RFC 6749, *The OAuth 2.0 Authorization Framework*, October 2012

185 <https://tools.ietf.org/html/rfc6749>

186 IETF RFC 6750, *The OAuth 2.0 Authorization Framework: Bearer Token Usage*, October 2012

187 <https://tools.ietf.org/html/rfc6750>

188 IETF RFC 8323, *CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets*,
189 February 2018

190 <https://tools.ietf.org/html/rfc8323>

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

193

194 **3 Terms, definitions, and abbreviated terms**

195 **3.1 Terms and definitions**

196 For the purposes of this document, the terms and definitions given in ISO/IEC 30118-1 and
197 ISO/IEC 30118-2 and the following apply.

198 ISO and IEC maintain terminological databases for use in standardization at the following
199 addresses:

200 – ISO Online browsing platform: available at <https://www.iso.org/obp>

201 – IEC Electropedia: available at <http://www.electropedia.org/>

202 **3.1.1**

203 **Cloud Provider**

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

205 **3.1.2**

206 **OCF Cloud**

207 logical entity that is owned by the *Cloud Provider* (3.1.1) that authorised to communicate with a
208 Device on behalf of the *OCF Cloud User* (3.1.3).

209 **3.1.3**

210 **OCF Cloud User**

211 Client that has permissions to interact with the Devices that are exposed by the *OCF Cloud* (3.1.2).

212 **3.1.4**

213 **Resource Directory**

214 set of descriptions of Resources where the actual Resources are held on Servers external to the
215 entity hosting the *Resource Directory* (3.1.4), allowing lookups to be performed for those Resources

216 **3.2 Symbols and abbreviated terms**

217 UX User Experience

218 **4 Document conventions and organization**

219 **4.1 Conventions**

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

224 In this document, to be consistent with the IETF usages for RESTful operations, the RESTful
225 operation words CRUDN, CREATE, RETRIVE, UPDATE, DELETE, and NOTIFY will have all letters
226 capitalized. Any lowercase uses of these words have the normal technical English meaning.

227 **4.2 Notation**

228 In this document, features are described as required, recommended, allowed or DEPRECATED as
229 follows:

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

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

234 Recommended (or should)(S).

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

241 Allowed (may or allowed)(O).

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

245 DEPRECATED.

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

252 Conditionally allowed (CA)

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

255 Conditionally required (CR)

256 – The definition or behaviour depends on a condition. If the specified condition is met, then the
257 definition or behaviour is required. Otherwise the definition or behaviour is allowed as default
258 unless specifically defined as not allowed.

259

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

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

262 5 Overview

263 5.1 Introduction

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

- 266 – CoAP over TCP protocol defined in ISO/IEC 30118-1
- 267 – The requirements within this document including those for a Resource Directory
- 268 – Security requirements and SVRs defined within the ISO/IEC 30118-2

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

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

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

280 5.2 OCF Cloud Architecture Alignment with ISO IEC 17789

281 Reference ISO/IEC 17789 defines a cloud computing reference architecture (CCRA) which can be
282 described in terms of one of four architectural viewpoints; user, functional, implementation, and
283 deployment. Of the four viewpoints, implementation and deployment are explicitly out of scope of
284 ISO/IEC 17789.

285 OCF defines an application capabilities type cloud service, providing Communication as a Service
286 (CaaS) (reference ISO/IEC 17788). This cloud service is provided by a cloud service provider, the
287 mechanisms used by the cloud service provider in managing their overall cloud infrastructure are
288 outside the scope of the OCF defined cloud service. The OCF definition is specific to the interface
289 offered by the cloud service to the cloud service customer, specifically the cloud service user.
290

291 There are three different user views defined. In the case where the cloud service customer is an
292 OCF Device as specified in this document then the views provided are:

- 293 - Interface for the OCF Device to provide information to the cloud service
- 294 - Interface for the OCF Device to retrieve information that has been provided to the cloud
295 service

296
297 In the case where the cloud service customer is another instance of a cloud service as specified in
298 OCF Cloud API for Cloud Services then the view provided is:

- 299 - Interface for the other cloud service instance to retrieve and update the information that is
300 provided via the cloud service

301
302 The OCF cloud service pertains specifically to a cloud service user, there is a single applicable
303 cloud service activity, that of "Use cloud service" defined in clause 8.2.21 of ISO/IEC 17789.

304
305 Credentials for the user of the cloud service are provided using OAuth2.0 as defined by
306 IETF RFC 6749. The cloud service, either itself, or leveraging an external authorization server,
307 provides a bearer token that is required in all requests from all cloud users. Please see clause 8.1
308 and OCF Cloud Security.
309

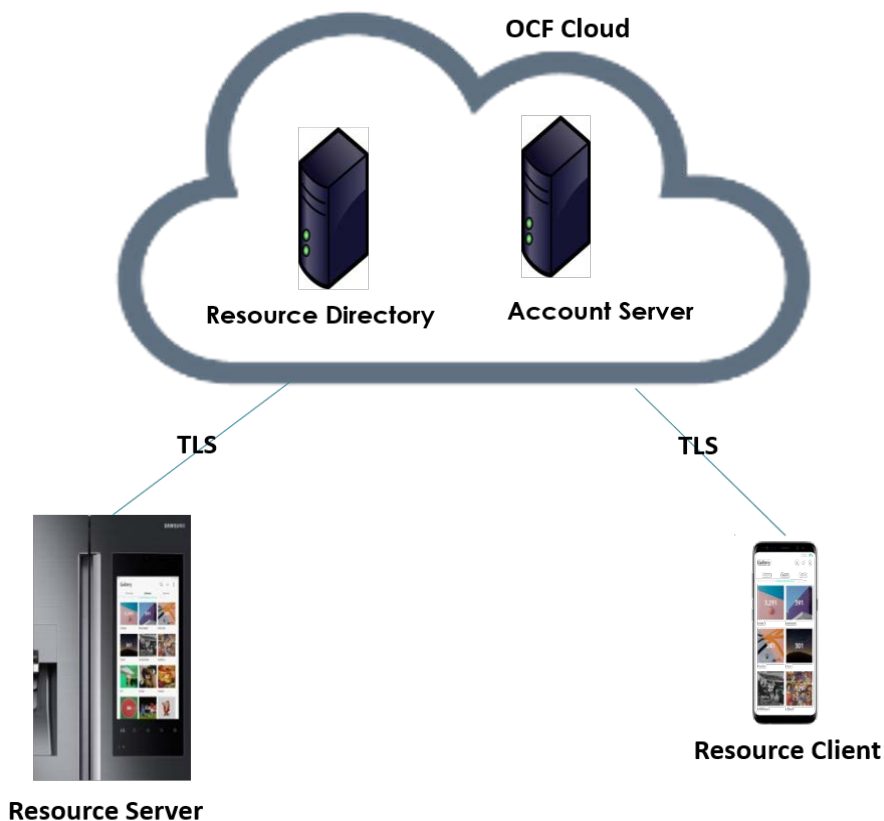
310 All connectivity between a cloud user and the cloud service is via mutually authenticated TLS; see
311 clause 7.1 of OCF Cloud Security.

312 **5.3 Architecture**

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

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

325 This is illustrated in Figure 1.



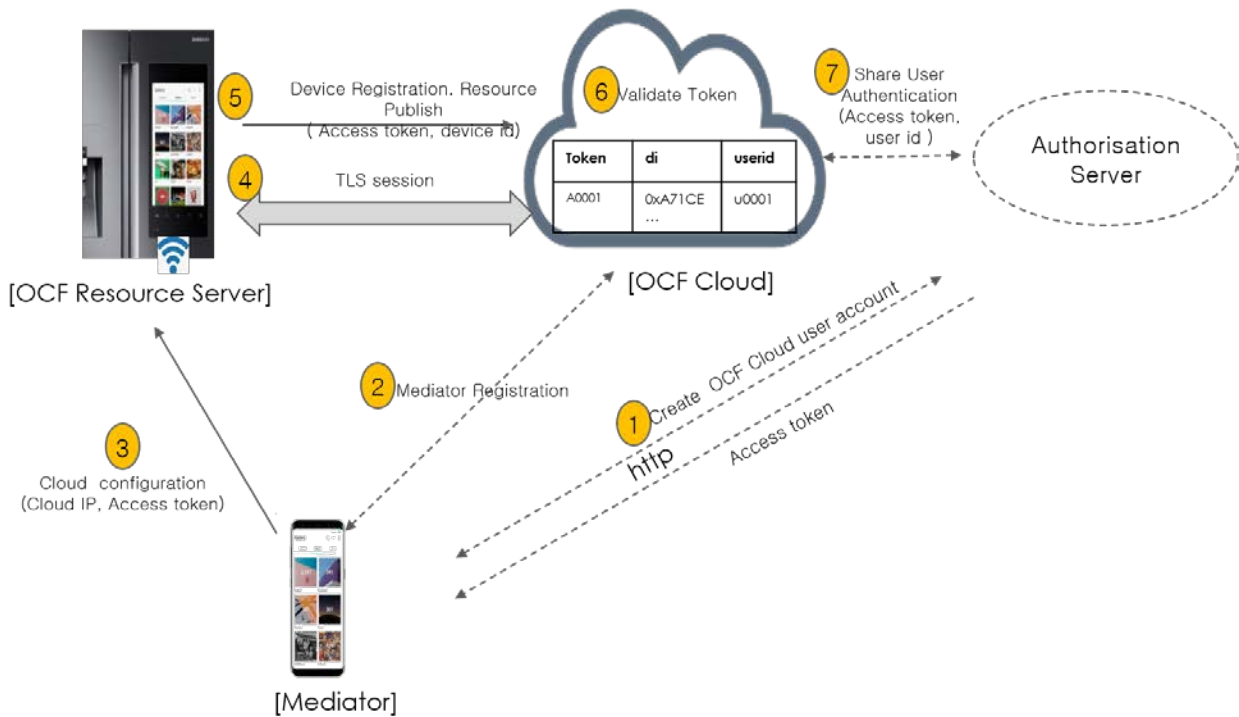
326

327

Figure 1 – OCF Cloud Architecture

328 **5.4 Interaction Flow**

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



331

332

Figure 2 – OCF Cloud interaction model

333

334

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.

335

336

337

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

338

5.5 Cloud Operational Flow

339

5.5.1 Introduction

340

341

342

343

The sub-clauses listed provide an informative overview of the flow which results on a Device being registered with an OCF Cloud and Client interaction with that Device. The clauses provide references to the applicable clauses within this document and other documents that provide normative details.

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

- 345 – Pre-requisites and OCF Cloud User account creation (see 5.5.2)
- 346 – Mediator registration with the OCF Cloud (see 5.5.3)
- 347 – Device provisioning by the Mediator (see 5.5.4)
- 348 – Device registration with the OCF Cloud (see 5.5.5)
- 349 – Device connection with the OCF Cloud (see 5.5.6)
- 350 – Devices Publishing Links to the OCF Cloud RD (see 5.5.7)
- 351 – Client to Server communication through the OCF Cloud (see 5.5.8)
- 352 – Device refreshing connection with the OCF Cloud (see 5.5.9)
- 353 – Device closing connection with the OCF Cloud (see 5.5.10)
- 354 – Device de-registering from the OCF Cloud (see 5.5.11)

355 **5.5.2 Pre-requisites and OCF Cloud User Account Creation**

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

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

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

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

365 **5.5.3 Mediator registration with the OCF Cloud**

366 See 8.1.2.2, 8.1.2.3.

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

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

374 **5.5.4 Device provisioning by the Mediator**

375 See 8.1.2.3; see also ISO/IEC 30118-2 clause 7.5.2

376 The Mediator connects to the Device through normal OCF processes. The Mediator then requests
377 an Access Token from the OCF Cloud for the Device being provisioned. The Mediator updates the
378 "oic.r.coapcloudconf" Resource on the Device with the Access Token received from the OCF Cloud,
379 the OCF Cloud URI, and the OCF Cloud UUID. The Mediator may also provide the Auth Provider
380 Name. Note that this Access Token may only be used one time for the initial Device Registration
381 with the OCF Cloud.

382 **5.5.5 Device Registration with the OCF Cloud.**

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

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

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

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

399 The Device is now registered with the OCF Cloud.

400 **5.5.6 Connection with the OCF Cloud**

401 See 8.1.4, see also ISO/IEC 30118-2 clause 13.12

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

406 **5.5.7 Publishing Links to the OCF Cloud RD**

407 See clauses 6.1.3.2 and 8.2; see also ISO/IEC 30118-2 clause 10.5.

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

410 **5.5.8 Client to Server communication through the OCF Cloud**

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

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

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

415 When the Client attempts CRUDN actions on the Links hosted by the OCF Cloud, the OCF Cloud
416 forwards those requests to the Device. The Device responds to the OCF Cloud which then proxies
417 the response to the Client (i.e. Client -> OCF Cloud -> Device -> OCF Cloud -> Client).

418 **5.5.9 Refreshing connection with the OCF Cloud**

419 See ISO/IEC 30118-2 clause 13.13.

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

422 **5.5.10 Closing connection with the OCF Cloud**

423 See ISO/IEC 30118-2 clause 13.12.

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

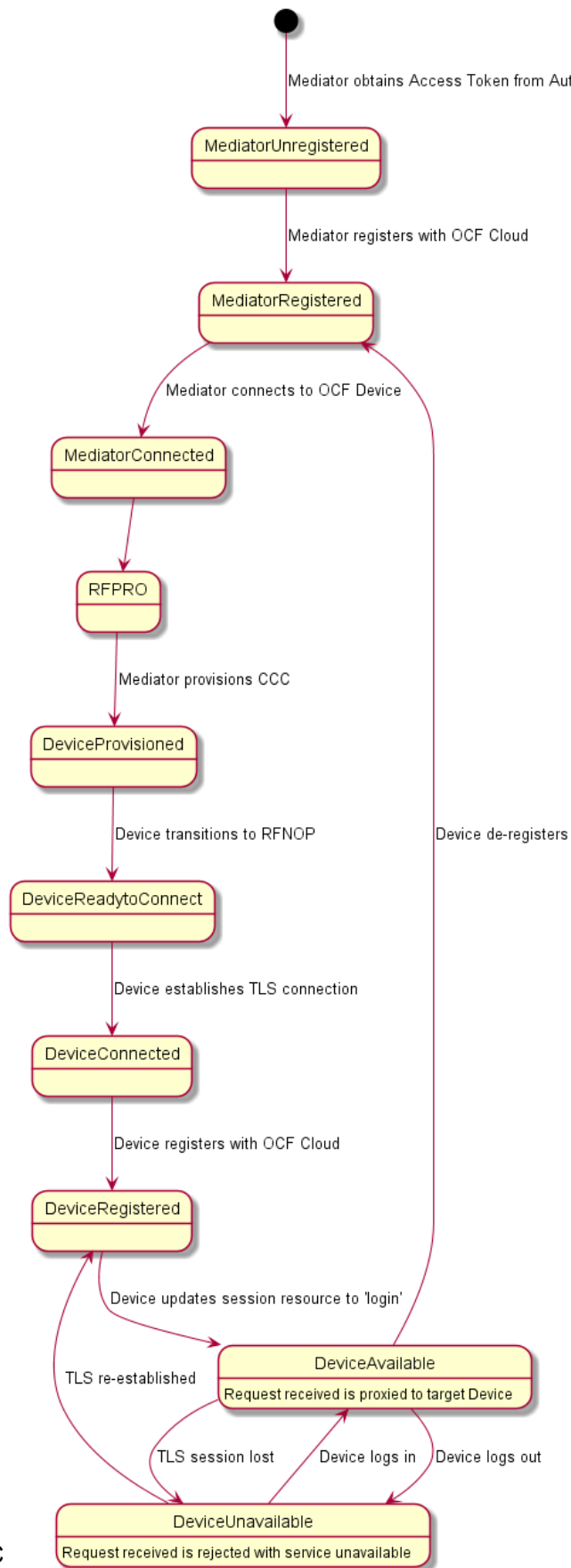
428 **5.5.11 Deregistering from the OCF Cloud**

429 See 8.5; see also ISO/IEC 30118-2 clause 13.10.

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

433 To connect to the OCF Cloud again, the Device has to re-follow the flow starting with Mediator
434 provisioning (see clause 5.5.4).

435 Figure 3 captures the state machine that is described by the informative operation flow provided in
436 clause 5.5.



438

Figure 3 – Overall Operational State Machine

439 6 Resource model

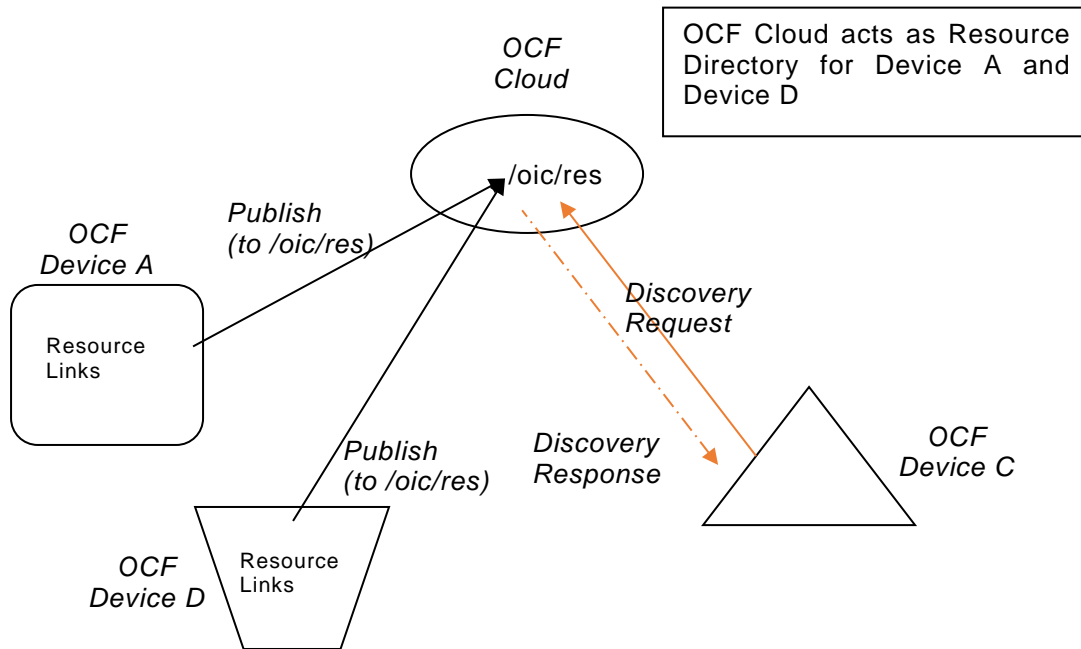
440 6.1 OCF Cloud Resource Directory

441 6.1.1 Indirect discovery for lookup of Resources

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

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

450



451

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

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

457 6.1.2 Resource Directory Definition

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

- 459 – *RD discovery* – the procedure by which publishing Devices discover an RD, in the case of the
460 OCF Cloud this is a direct result of Device registration with an OCF Cloud.
- 461 – *Resource publish* – the procedures with which Devices publish their Resource information, i.e.
462 Links.

463 – *Resource exposure* – the feature with which RDs expose the Links hosted by the 3rd party
 464 Devices via their own "/oic/res".

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

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

474 **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

475

476 **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.

477

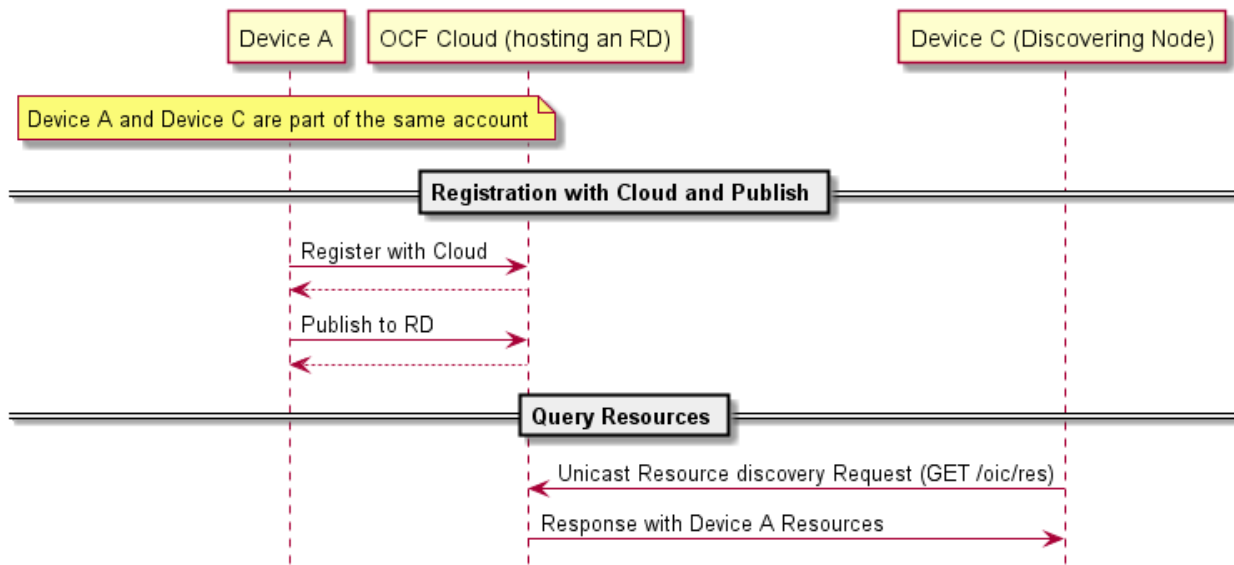
478 An RD may be queried at its "/oic/res" Resource to find Resources hosted on other Devices. A
 479 publishing Device may publish all or a partial list of Resources they host to an RD. The RD then
 480 responds to queries for Resource discovery on behalf of the publishing Device. Note that only
 481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

484 **6.1.3 RD operational flows**

485 **6.1.3.1 Discovering an RD**

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

488



489
490 **Figure 5 – RD discovery and RD supported query of Resources support**

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

494 **6.1.3.2 Publish Resources**

495 **6.1.3.2.1 Overview**

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

498 The publishing Device shall mark as observable all Resources that are to be published to the RD,
499 see clause 11.3.2 of ISO/IEC 30118-1. The minimum set of Resources that a publishing Device
500 shall publish are the mandatory Core Resources "/oic/d" and "/oic/p" as well as Resources that are
501 defined as mandatory for the Device Type being published. The publishing Device may publish
502 additional Resources beyond the mandatory set identified in this clause. The publishing Device
503 should only publish Resources that are otherwise published to its own "/oic/res"; a publishing
504 Device should not publish non-Discoverable Resources or Resources hosted by some other Device.

505 A publishing Device shall respond to discovery requests on its "/oic/res" Resource unless all its
506 Discoverable Resources have been published in an RD.

507 **6.1.3.2.2 Publish: Push Resource information**

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

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

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

- 514 – "di" –its value shall be the Device UUID of the publishing Device, i.e. the "di" value of "/oic/d".
- 515 – "links" –its value shall be the array of Links to be published. Links may omit the "ins" Parameter
- 516 in which case the RD will assign a value for each Link. The supplied "ins" Parameter by the
- 517 Client is allowed to be overruled by the RD, e.g. an RD can ignore the supplied "ins" value.

518 – "ttl" –its value indicates how long (in seconds) the publishing Device requests the RD to keep
519 this published Link.

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

```
521 {  
522   "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",  
523   "links": [  
524     {  
525       "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9"  
526       "href": "/myLightSwitch",  
527       "rt": ["oic.r.switch.binary"],  
528       "if": ["oic.if.a", "oic.if.baseline"],  
529       "p": {"bm": 3},  
530       "eps": [  
531         {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2},  
532         {"ep": "coaps://[fe80::b1d6]:1122"},  
533         {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}  
534       ]  
535     },  
536     {  
537       "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",  
538       "href": "/myLightBrightness",  
539       "rt": ["oic.r.brightness"],  
540       "if": ["oic.if.a", "oic.if.baseline"],  
541       "p": {"bm": 3},  
542       "eps": [  
543         {"ep": "coaps://[[2001:db8:a::123]:2222"}  
544       ]  
545     }  
546   ],  
547   "ttl": 600  
548 }
```

549 When an RD receives this initial UPDATE request, it determines whether to grant the request or
550 not. If the UPDATE request includes any Links that are not marked as observable, then the request
551 is not granted, and the RD shall reject that request with an error response (e.g. "Bad Request"). If
552 the request is granted, the RD shall send back a success path UPDATE response to the publishing
553 Device. The response shall include a payload with the same information as the original UPDATE
554 request with the following possible differences:

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

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

```
566 {  
567   "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",  
568   "links": [  
569     {  
570       "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",  
571       "href": "/myLightSwitch",  
572       "rt": ["oic.r.switch.binary"],  
573       "if": ["oic.if.a", "oic.if.baseline"],
```

```

574     "p": {"bm": 3},
575     "eps": [
576         {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2},
577         {"ep": "coaps://[fe80::b1d6]:1122"},
578         {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}
579     ],
580     "ins": 11235
581 },
582 {
583     "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
584     "href": "/myLightBrightness",
585     "rt": ["oic.r.brightness"],
586     "if": ["oic.if.a", "oic.if.baseline"],
587     "p": {"bm": 3},
588     "eps": [
589         {"ep": "coaps://[[2001:db8:a::123]:2222"}
590     ],
591     "ins": 112358
592 }
593 ].
594 "ttl": 600
595 }

```

596 6.1.3.3 Resource exposure

597 6.1.3.3.1 "/oic/res" and retrieving of the Resources

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

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

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

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

```

611 [
612   {
613     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
614     "href": "/oic/res",
615     "rel": "self",
616     "rt": ["oic.wk.res"],
617     "if": ["oic.if.ll", "oic.if.baseline"],
618     "p": {"bm": 3},
619     "eps": [
620         {"ep": "coap://[2001:db8:a::b1d4]:77777"},
621         {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
622     ]
623   },
624   {
625     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
626     "href": "/oic/d",
627     "rt": ["oic.wk.d", "oic.d.fan"],
628     "if": ["oic.if.r", "oic.if.baseline"],
629     "p": {"bm": 3},

```

```

630     "eps": [
631         {"ep": "coap://[2001:db8:a::b1d4]:77777"},
632         {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
633     ]
634 },
635 {
636     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
637     "href": "/oic/p",
638     "rt": ["oic.wk.p"],
639     "if": ["oic.if.r", "oic.if.baseline"],
640     "p": {"bm": 3},
641     "eps": [
642         {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
643     ]
644 },
645 {
646     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
647     "href": "/oic/rd",
648     "rt": ["oic.wk.rd"],
649     "if": ["oic.if.baseline"],
650     "p": {"bm": 3},
651     "eps": [
652         {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
653     ]
654 },
655 {
656     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
657     "href": "/myFanSwitch",
658     "rt": ["oic.r.switch.binary"],
659     "if": ["oic.if.a", "oic.if.baseline"],
660     "p": {"bm": 3},
661     "eps": [
662         {"ep": "coaps://[2001:db8:a::b1d4]:33333"}
663     ]
664 },
665 {
666     "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
667     "href": "/oic/d",
668     "rt": ["oic.wk.d", "oic.d.light"],
669     "if": ["oic.if.r", "oic.if.baseline"],
670     "p": {"bm": 3},
671     "eps": [
672         {"ep": "coap://[2001:db8:b::c2e5]:66666"},
673         {"ep": "coaps://[2001:db8:b::c2e5]:22222"}
674     ]
675 },
676 {
677     "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
678     "href": "/oic/p",
679     "rt": ["oic.wk.p"],
680     "if": ["oic.if.r", "oic.if.baseline"],
681     "p": {"bm": 3},
682     "eps": [
683         {"ep": "coaps://[2001:db8:b::c2e5]:22222"}
684     ]
685 },
686 {
687     "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
688     "href": "/myLightSwitch",
689     "rt": ["oic.r.switch.binary"],
690     "if": ["oic.if.a", "oic.if.baseline"],
691     "p": {"bm": 3},
692     "eps": [

```



```

693     {"ep": "coaps://[2001:db8:b::c2e5]:22222"}
694   ]
695 },
696 {
697   "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
698   "href": "/myLightBrightness",
699   "rt": ["oic.r.brightness"],
700   "if": ["oic.if.a", "oic.if.baseline"],
701   "p": {"bm": 3},
702   "eps": [
703     {"ep": "coaps://[2001:db8:b::c2e5]:22222"}
704   ]
705 }
706 ]

```

707

708 6.2 CoAPCloudConf Resource

709 6.2.1 Introduction

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

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

716 6.2.2 Resource Definition

717 The CoAPCloudConf Resource is as defined in Table 4.

718

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

719

720

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

722 **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 ¹	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
Cloud Provisioning Status	"cps"	string	enum	N/A	R	No	Cloud provisioning status of Device. One of: "uninitialized", "readytoregister", "registering", "registered", "failed"

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

723
724 If the "clec" Property is implemented by a Device, it shall have an initial value of 0 ("No error").

725 **6.2.3 Cloud status governing state machine**

726 **6.2.3.1 Introduction**

727 The "cps" Property exposes the registration state of the Device with an OCF Cloud. The states supported are listed in Table 6.

729

730

Table 6 – Device registration states

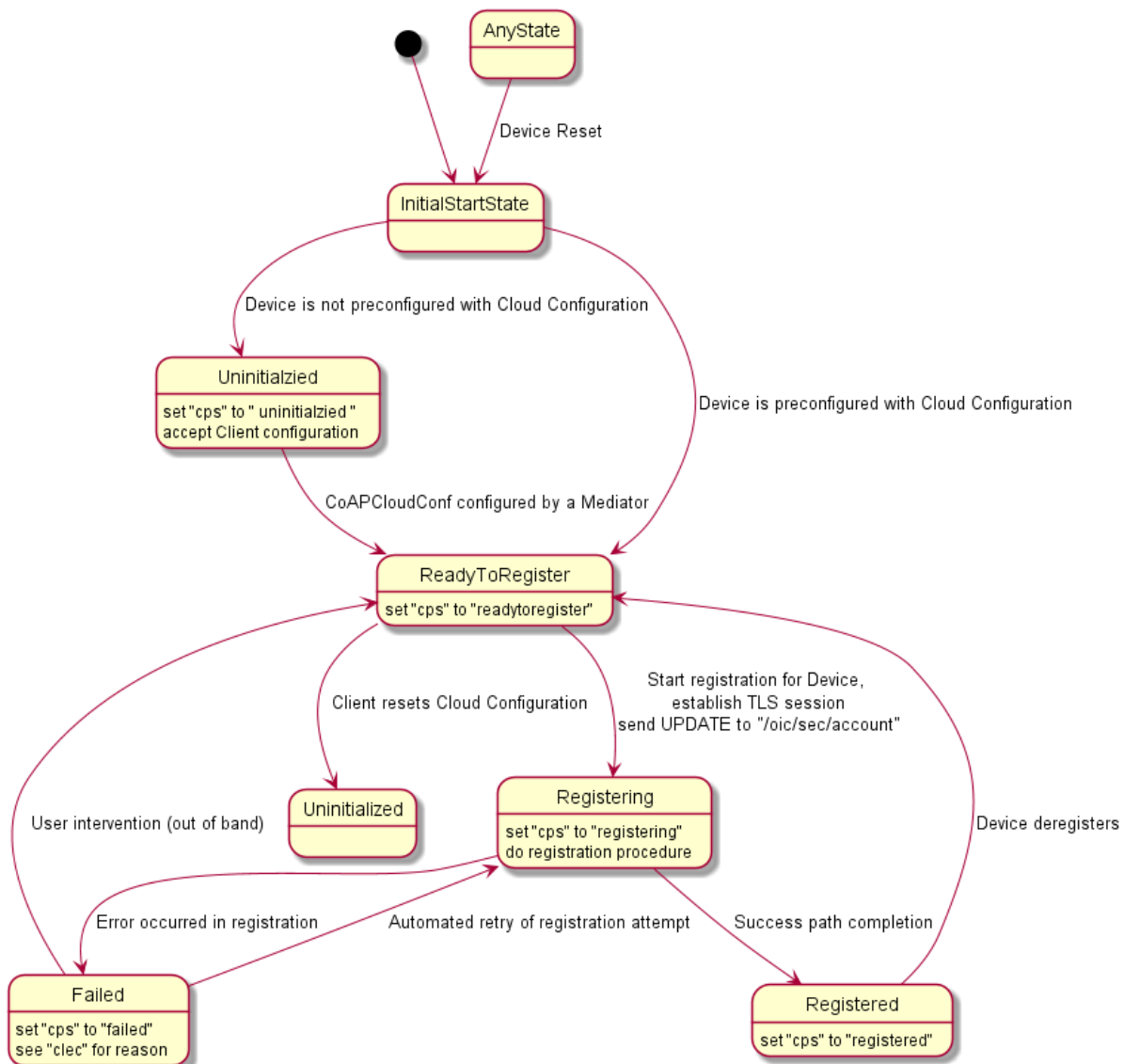
State	Description
"uninitialized"	Device is not initialized (i.e. CoAPCloudConf Properties set) with information of the OCF Cloud to which it will connect.
"readytoregister"	Device has been configured but not registered with the target OCF Cloud.
"registering"	A TLS session is being established, or a TLS session has been established and the Device has sent an UPDATE operation to "/oic/sec/account" as defined in clause 8.1.4 and is waiting on a response.
"registered"	The Device has received a success path response from the UPDATE operation to "/oic/sec/account".
"failed"	The Device experiences a failure during Cloud Provisioning, e.g. the Device does not receive a Success path response from the UPDATE operation. The "clec" Property when in the "failed" state if exposed indicates the specific failure reason.

731

732

733

Figure 6 details the state machine which describes the transitions between the values that are exposed by the "cps" Property.



734
735
736 **Figure 6 – Device registration status state machine**

737 **6.2.3.2 State definitions**

738 **6.2.3.2.1 "uninitialized" state**

739 The Device has not been configured by a Mediator with resolvable information for the "cis", "sid",
740 or "at" Properties of the "oic.r.coapcloudconf" Resource Type (i.e. the "cis" is a URI that cannot be
741 resolved, and the "sid" is a null UUID). A Device may be in this state as an initial state. A Device
742 shall transition into this state as a result of a Device reset (an appropriately privileged Client or
743 OBT setting of "pstat") if there is no pre-configured information. It shall not be possible to perform
744 an UPDATE operation to modify the Properties of the CoAPCloudConf Resource in any state other
745 than "uninitialized", "readytoregister" or "failed" states.

746 **6.2.3.2.2 "readytoregister" state**

747 The Device has been configured by a Mediator with information for the "cis", "sid", and "at"
748 Properties of the "oic.r.coapcloudconf" Resource Type, but has no connectivity to the OCF Cloud
749 and is not in the process of establishing such connectivity. A Device may be in this state as an

750 initial state. The Device shall transition to this state from the "uninitialized" state once it has been
751 configured with values for the "cis", "at", and "sid" Properties in "oic.r.coapcloudconf". by a Mediator.
752 A Device shall transition into this state as a result of a Device reset (Client setting of the "pstat"
753 Property) if there is pre-configured information.

754 **6.2.3.2.3 "registering" state**

755 The Device shall transition to "registering" once the TLS handshake to the OCF Cloud is initiated.
756 The Device shall transition from "registering" to "registered" on reception of a success path
757 response to the UPDATE operation sent to the "/oic/sec/account" Resource as defined in clause
758 8.1.4. If a non-success path response is received to the UPDATE operation sent to the
759 "/oic/sec/account" Resource the Device shall transition to the "failed" state, unless the Device
760 autonomously re-attempts the registration by sending an UPDATE operation to the
761 "/oic/sec/account" Resource as defined in clause 8.1.4. In this latter instance the Device shall
762 remain in the "registering" state.

763 **6.2.3.2.4 "registered" state**

764 The Device has completed registration with the OCF Cloud as defined in clause 8.1.4. If the Device
765 subsequently deregisters in accordance with clause 8.5 the Device shall transition to the
766 "readytoregister" state.

767 **6.2.3.2.5 "failed" state**

768 The Device has received a non-success path response from the OCF Cloud during the registration
769 procedure as defined in clause 8.1.4 and is not attempting an autonomous retry or re-attempt. The
770 Device may offer some out of band means, or user intervention scheme, that allows the transition
771 from the "failed" state to the "readytoregister" or the "uninitialized" state to enable re-attempt.

772 The "clec" Property, if exposed, shall be populated with the specific failure reason why the Device
773 is in the "failed" state.

774 **6.2.4 Error Handling**

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

- 779 – The Device shall set the "clec" Property to 1 if it receives an error response from the OCF Cloud
780 (e.g. error response from the Cloud).
- 781 – The Device shall set the "clec" Property to 2 if there is a failure to connect to the OCF Cloud
782 (e.g. no reply, timeout, or timeout).
- 783 – The Device shall set the "clec" Property to 3 if it fails to refresh the Access Token (e.g. if it
784 receives an error response during the token refresh procedure).

785 **7 Network and connectivity**

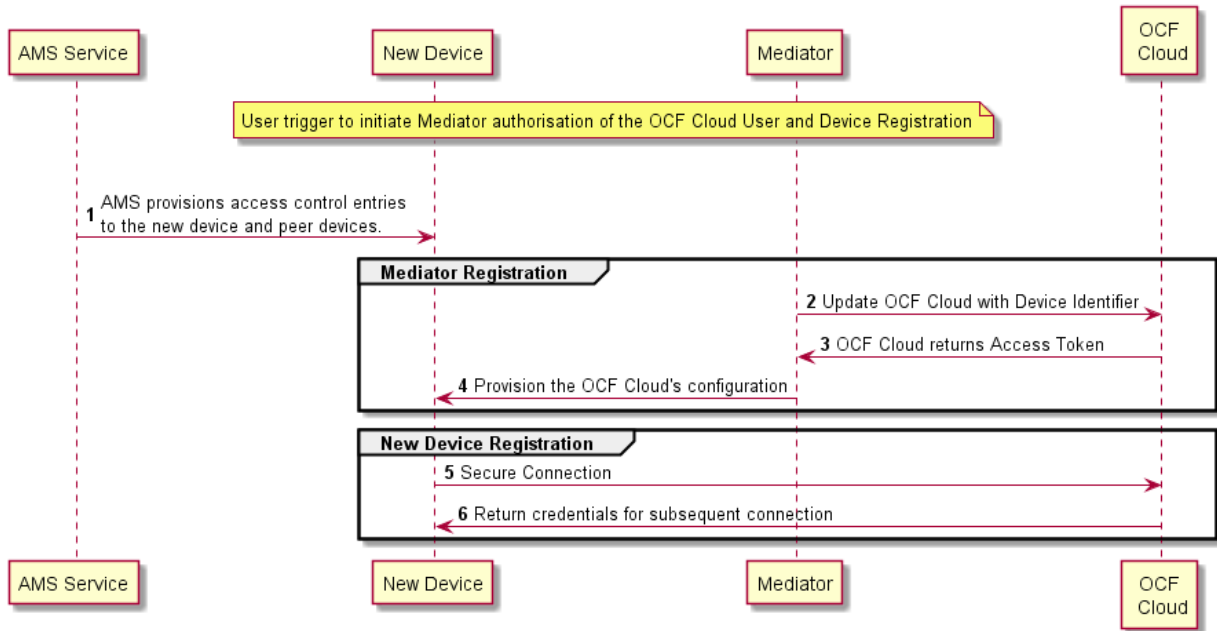
786 A TLS session exists between a Device and the OCF Cloud as specified in IETF RFC 8323; this is
787 established following device configuration as detailed in 8.1.2.3.

788 **8 Functional interactions**

789 **8.1 Onboarding, Provisioning, and Configuration**

790 **8.1.1 Overview**

791 Figure 7 provides an overview of the interaction between the different entities to get the Device
 792 registered with the OCF Cloud. A summary of the flow is provided in Table 4.



793 **Figure 7 – Registration with OCF Cloud**

794 **Table 7 – Device to OCF Cloud Registration Flow**

795

796

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.

797

798 **8.1.2 Use of Mediator**

799 **8.1.2.1 Introduction**

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

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

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

- 809 – OCF Cloud Interface URL ("cis") Property
- 810 – OCF Cloud UUID ("sid") Property (to verify Cloud identity)
- 811 – Access Token ("at") Property that is validated by the OCF Cloud
- 812 – Optionally the Authorisation Provider name ("apn") Property through which the Access Token
813 was obtained

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

817 **8.1.2.2 OCF Cloud User Authorisation of the Mediator**

818 The Mediator uses a user authorisation mechanism to enable the OCF Cloud to validate the OCF
819 Cloud User's authorisation and obtain the OCF Cloud User's identity. The Authorisation Provider
820 should be trusted by both the OCF Cloud User and the OCF Cloud. The Mediator may use OAUTH
821 2.0 (see IETF RFC 6749) or another user authentication mechanism to obtain an Access Token as
822 a form of authorisation from an OCF Cloud User via an Authorisation Provider. This authorisation
823 achieves a variety of purposes. Firstly, the authorisation shows OCF Cloud User consent for
824 Mediator to connect to the OCF Cloud. Secondly, the authorisation is used to obtain information to
825 map the Devices to the same OCF Cloud User.

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

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

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

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

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

837 The OCF Cloud then verifies the Access Token with the Authorisation Provider. If the Authorisation
838 Provider validates the Access Token successfully, then it will return information about the OCF
839 Cloud User to whom the Access Token belongs. The OCF Cloud generates a unique Access Token
840 for the Mediator (which may be the original Access Token from the Mediator or a new Access Token)
841 and a User ID (i.e. "uid" Property of "oic.r.account") if this is the first instance of registering a
842 Mediator with this OCF Cloud User. The User ID acts as a unique identity for the OCF Cloud User.
843 All instances of a Mediator for the same OCF Cloud User will be associated with the same User ID.
844 This information is returned to the Mediator over TLS. The returned Access Token and User ID are
845 used by the OCF Cloud to identify the Mediator. This returned Access Token is used by the
846 Mediator in subsequent interactions with the OCF Cloud.

847 All Devices registering with the OCF Cloud receive the same User ID from the OCF Cloud when
848 registering with the same Mediator.

849 **8.1.2.3 Device Provisioning by the Mediator**

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

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

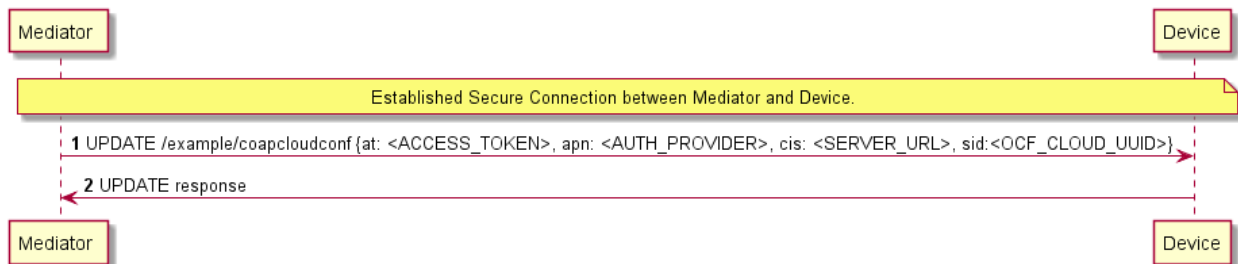
- 858 – Device UUID (i.e. "di" Property Value of "/oic/d" of the Device)

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

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

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

875 Figure 8 describes the flow for provisioning of the Device by a Mediator. Table 8 provides additional context around the flow.



877

878

Figure 8 – Device Provisioning by the Mediator

879

880

Table 8 – 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

881

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

884 **8.1.3 Device Connection to the OCF Cloud**

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

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

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

899 **8.1.4 Device Registration with the OCF Cloud**

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

904 After the TLS connection is established with the OCF Cloud, the Device shall register with the OCF
905 Cloud by sending an UPDATE request to "/oic/sec/account" as defined in clause 13.10 of the
906 ISO/IEC 30118-2. The OCF Cloud consequently associates the TLS connection with the
907 corresponding "uid" and "di" Properties populated in the "/oic/sec/account/" Resource. Any other
908 Device registering with the OCF Cloud is assigned the same User ID by the OCF Cloud when
909 registering with any Mediator associated with that User ID. Device Registration permits a Client to
910 access Resources on the OCF Cloud which are associated with the same User ID as the Client.

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

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

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

920 On completion of Device Registration the Device shall send an UPDATE to "/oic/sec/session" as
921 defined in clause 13.11 of the ISO/IEC 30118-2 to ensure that the established TLS session is
922 maintained for subsequent interaction with the OCF Cloud Resource Directory as defined in clause
923 8.2.

924 **8.2 Resource Publication**

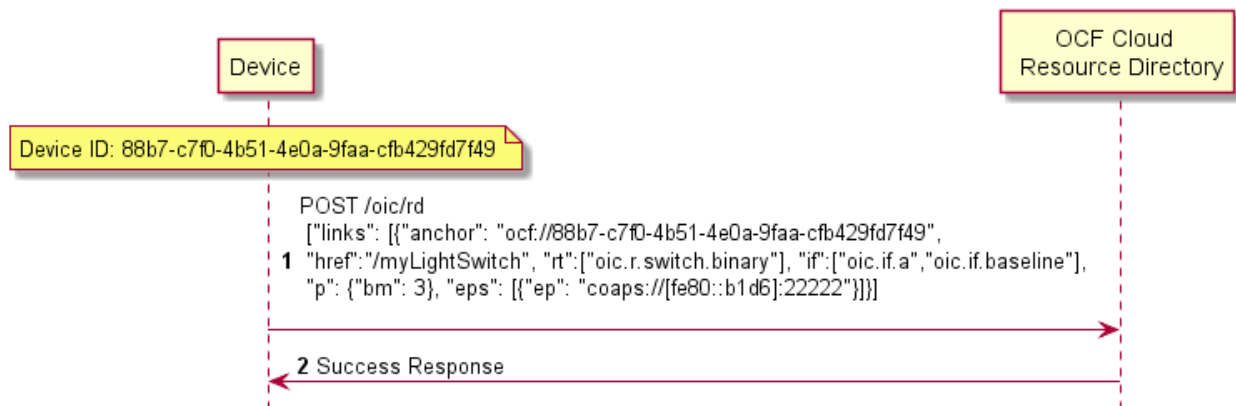
925 An OCF Cloud exposes a Resource Directory as defined in clause 6.1. After a Device is registered
926 with an OCF Cloud, the Device should publish its Resources to the OCF Cloud's Resource Directory
927 following the procedures defined in clause 6.1.3.2. The Device and OCF Cloud maintain a

928 persistent TLS connection over which requests received by the OCF Cloud for the Device are
929 routed.

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

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

944 Figure 9 provides an example showing the provided Device UUID from the Device; Figure 10 shows
945 the pre-pending of the Device UUID to the "href" Link Parameter in the Resource Directory itself.



946

947

Figure 9 – Resource publication to the OCF Cloud

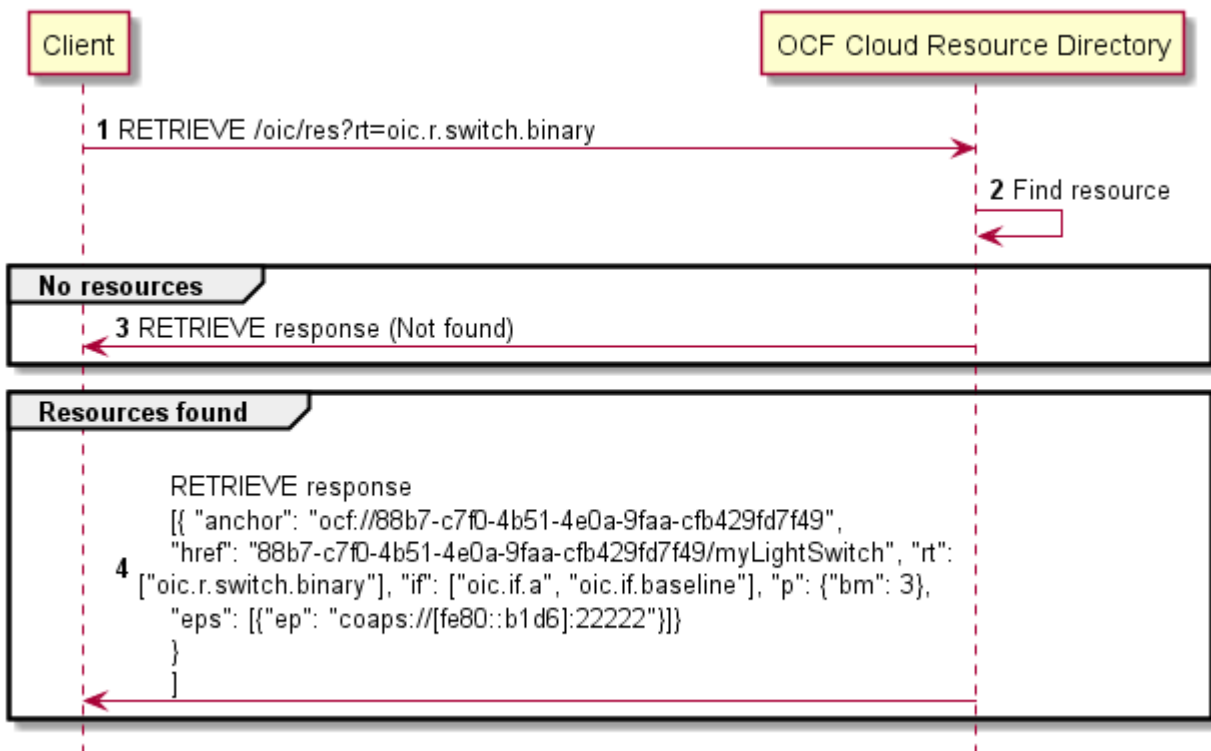
948 8.3 Client Registration with the OCF Cloud

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

952 8.4 Resource Discovery

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

958 Figure 10 provides an illustrative flow for Resource Discovery, note the population of the 'href' for
959 instance of "oic.r.switch.binary" including the Device UUID of the target Device in accordance with
960 8.2:



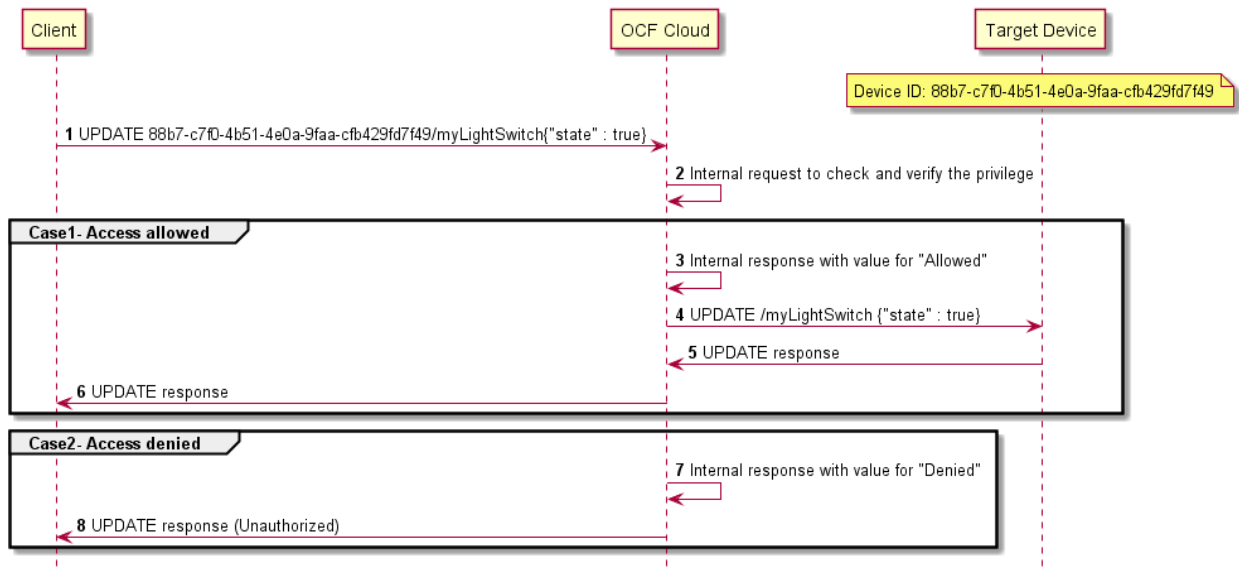
961

962

Figure 10 – Resource discovery through OCF Cloud

963 The OCF Cloud acts as a simple proxy, forwarding the messages to the publishing Devices. The
 964 remote Device sends a RETRIEVE to the OCF Cloud to obtain the content of the Server's published
 965 Resources, the OCF Cloud will route the message to the target Device after first removing the
 966 Device UUID that had been prepended to the 'href' Link Parameter by the Cloud RD. Similarly,
 967 other CRUDN operations originated by a Client are routed to the Server via the OCF Cloud. The
 968 publishing Device treats the forwarded request message as a request from the OCF Cloud. The
 969 publishing Device authorises the request as specified in ISO/IEC 30118-2, using the UUID of the
 970 OCF Cloud configured in the "sid" Property of "oic.r.coapcloudconf". The publishing Device sends
 971 a response message to the OCF Cloud, and the OCF Cloud forwards the response to the Client
 972 which sent the corresponding request.

973 Figure 11 illustrates request routing via the OCF Cloud



974
975 **Figure 11 – Request routing through OCF Cloud**

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

978 **8.5 Device Deregistration from the OCF Cloud**

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

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

983 **8.6 Device Management**

984 **8.6.1 Behaviours on Device maintenance state changes**

985 The OCF Core Optional Framework details actions on Device state transitions. This clause defines
986 the actions to be taken for the functionality defined within this document.

987 Table 9 provides a summary of the actions to be taken.

988 **Table 9 – Actions on Device state change**

	Soft reset	Hard reset	RFNOP -> RFPRO	RFPRO -> RFNOP
OCF Cloud	No change	See this clause	No change	No change

989 On a hard reset the Device, if registered to an OCF Cloud, shall de-register from the OCF Cloud in
990 accordance with the procedures in the ISO/IEC 30118-2, clause 13.10.

991 Further, on a hard reset the CoAPCloudConf Resource ("oic.r.coapcloudconf") shall be modified in
992 accordance with Table 10 for those Properties that are implemented.

993 **Table 10 – Default values for CoAPCloudConf Resource**

Property	Default	Notes
"apn"	""	Empty string, only if no manufacturer default exists, in which case it reverts to that default or is unchanged.

"cis"	"coaps+tcp://127.0.0.1"	Or other valid but non-resolving URI.
"at"	""	Empty string, only if no manufacturer default exists, in which case it reverts to that default or is unchanged.
"sid"	Temporary not repeated value or "00000000-0000-0000-0000- 000000000000"	
"clec"	0	No error.

994

995 **9 Security**

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

997

Annex A (normative)

Swagger2.0 definitions

A.1 List of Resource Type definitions

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

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

A.2 Resource directory resource

A.2.1 Introduction

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

A.2.2 Well-known URI

/oic/rd

A.2.3 Resource type

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

A.2.4 OpenAPI 2.0 definition

```
{
  "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": {
          "200": {
            "description": "Respond with the selector criteria - either the set of attributes or
the bias factor\n",
            "x-example": {
```

```

1046         "rt": ["oic.wk.rd"],
1047         "if": ["oic.if.baseline"],
1048         "sel": 50
1049     },
1050     "schema": { "$ref": "#/definitions/rdSelection" }
1051 }
1052 }
1053 },
1054 "post": {
1055     "description": "Publish the Resource information for the first time in /oic/res. Updates to
1056 existing entries are not allowed.\nAppropriates parts of the information, i.e., Links of the
1057 published Resources will be discovered through /oic/res.\n1) When a Device first publishes a Link,
1058 the request payload to RD may include the Links without an \"ins\" Parameter.\n2) Upon granting the
1059 request, the RD assigns a unique instance value identifying the Link among all the Links it
1060 advertises\n and sends back the instance value in the \"ins\" Parameter in the Link to the
1061 publishing Device.\n",
1062     "parameters": [
1063         { "$ref": "#/parameters/rdpostinterface" },
1064         {
1065             "name": "body",
1066             "in": "body",
1067             "required": true,
1068             "schema": { "$ref": "#/definitions/rdPublish" },
1069             "x-example": {
1070                 "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
1071                 "links": [
1072                     {
1073                         "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
1074                         "href": "/myLightSwitch",
1075                         "rt": [ "oic.r.switch.binary" ],
1076                         "if": [ "oic.if.a", "oic.if.baseline" ],
1077                         "p": { "bm": 3 },
1078                         "eps": [
1079                             { "ep": "coaps://[2001:db8:a::b1d6]:1111", "pri": 2 },
1080                             { "ep": "coaps://[2001:db8:a::b1d6]:1122" },
1081                             { "ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3 }
1082                         ]
1083                     },
1084                     {
1085                         "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
1086                         "href": "/myLightBrightness",
1087                         "rt": [ "oic.r.brightness" ],
1088                         "if": [ "oic.if.a", "oic.if.baseline" ],
1089                         "p": { "bm": 3 },
1090                         "eps": [
1091                             { "ep": "coaps://[2001:db8:a::123]:2222" }
1092                         ]
1093                     }
1094                 ],
1095                 "ttl": 600
1096             }
1097         }
1098     ],
1099     "responses": {
1100         "200": {
1101             "description": "Respond with the same schema as publish with the additional \"ins\"
1102 Parameter in the Link.\n",
1103             "x-example": {
1104                 "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
1105                 "links": [
1106                     {
1107                         "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
1108                         "href": "/myLightSwitch",
1109                         "rt": [ "oic.r.switch.binary" ],
1110                         "if": [ "oic.if.a", "oic.if.baseline" ],
1111                         "p": { "bm": 3 },
1112                         "eps": [
1113                             { "ep": "coaps://[2001:db8:a::b1d6]:1111", "pri": 2 },
1114                             { "ep": "coaps://[2001:db8:a::b1d6]:1122" },
1115                             { "ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3 }
1116                         ]
1117                     }
1118                 ],

```

```

1117         "ins": 11235
1118     },
1119     {
1120         "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
1121         "href": "/myLightBrightness",
1122         "rt": ["oic.r.brightness"],
1123         "if": ["oic.if.a", "oic.if.baseline"],
1124         "p": {"bm": 3},
1125         "eps": [
1126             {"ep": "coaps://[2001:db8:a::123]:2222"}
1127         ],
1128         "ins": 112358
1129     }
1130 ],
1131 "ttl": 600
1132 },
1133 "schema": { "$ref": "#/definitions/rdPublish" }
1134 }
1135 }
1136 }
1137 }
1138 },
1139 "parameters": {
1140     "rdgetinterface" : {
1141         "in" : "query",
1142         "name" : "if",
1143         "type" : "string",
1144         "enum" : ["oic.if.baseline"]
1145     },
1146     "rdpostinterface" : {
1147         "in" : "query",
1148         "name" : "if",
1149         "type" : "string",
1150         "enum" : ["oic.if.baseline"]
1151     }
1152 },
1153 "definitions": {
1154     "rdSelection" : {
1155         "properties": {
1156             "rt" : {
1157                 "description": "Resource Type of the Resource",
1158                 "items": {
1159                     "enum": ["oic.wk.rd"],
1160                     "type": "string",
1161                     "maxLength": 64
1162                 },
1163                 "minItems": 1,
1164                 "uniqueItems": true,
1165                 "readOnly": true,
1166                 "type": "array"
1167             },
1168             "n" : {
1169                 "$ref":
1170 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1171 schema.json#/definitions/n"
1172             },
1173             "sel" : {
1174                 "description": "A bias factor calculated by the Resource Directory",
1175                 "maximum": 100,
1176                 "minimum": 0,
1177                 "readOnly": true,
1178                 "type": "integer"
1179             },
1180             "id" : {
1181                 "$ref":
1182 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1183 schema.json#/definitions/id"
1184             },
1185             "if" : {
1186                 "description": "The OCF Interfaces supported by this Resource",
1187                 "items": {

```



```

1188         "enum": [
1189             "oic.if.baseline"
1190         ],
1191         "type": "string",
1192         "maxLength": 64
1193     },
1194     "minItems": 1,
1195     "readOnly": true,
1196     "uniqueItems": true,
1197     "type": "array"
1198 }
1199 },
1200 "type" : "object",
1201 "required": ["sel"]
1202 },
1203 "rdPublish" : {
1204     "properties": {
1205         "di" : {
1206             "$ref":
1207 "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1208 schema.json#/definitions/di"
1209         },
1210         "ttl" : {
1211             "description": "Time to indicate a RD, i.e. how long to keep this published item.",
1212             "type": "integer"
1213         },
1214         "links" : {
1215             "description": "A set of simple or individual OCF Links.",
1216             "items": {
1217                 "properties": {
1218                     "anchor": {
1219                         "$ref":
1220 "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1221 schema.json#/definitions/anchor"
1222                     },
1223                     "di": {
1224                         "$ref":
1225 "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1226 schema.json#/definitions/di"
1227                     },
1228                     "eps": {
1229                         "$ref":
1230 "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1231 schema.json#/definitions/eps"
1232                     },
1233                     "href": {
1234                         "$ref":
1235 "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1236 schema.json#/definitions/href"
1237                     },
1238                     "if": {
1239                         "description": "The interface set supported by the published resource",
1240                         "items": {
1241                             "enum": [
1242                                 "oic.if.baseline",
1243                                 "oic.if.ll",
1244                                 "oic.if.b",
1245                                 "oic.if.rw",
1246                                 "oic.if.r",
1247                                 "oic.if.a",
1248                                 "oic.if.s"
1249                             ],
1250                             "type": "string",
1251                             "maxLength": 64
1252                         },
1253                         "minItems": 1,
1254                         "uniqueItems": true,
1255                         "type": "array"
1256                     },
1257                     "ins": {
1258                         "$ref":

```

```

1259 "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1260 schema.json#/definitions/ins"
1261     },
1262     "p": {
1263         "$ref":
1264         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1265         schema.json#/definitions/p"
1266     },
1267     "rel": {
1268         "description": "The relation of the target URI referenced by the Link to the context
1269 URI",
1270         "oneOf": [
1271             {
1272                 "default": [
1273                     "hosts"
1274                 ],
1275                 "items": {
1276                     "maxLength": 64,
1277                     "type": "string"
1278                 },
1279                 "minItems": 1,
1280                 "type": "array"
1281             },
1282             {
1283                 "default": "hosts",
1284                 "maxLength": 64,
1285                 "type": "string"
1286             }
1287         ]
1288     },
1289     "rt": {
1290         "description": "Resource Type of the published Resource",
1291         "items": {
1292             "maxLength": 64,
1293             "type": "string"
1294         },
1295         "minItems": 1,
1296         "maxItems": 1,
1297         "uniqueItems": true,
1298         "type": "array"
1299     },
1300     "title": {
1301         "$ref":
1302         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1303         schema.json#/definitions/title"
1304     },
1305     "type": {
1306         "$ref":
1307         "https://openconnectivityfoundation.github.io/core/schemas/oic.links.properties.core-
1308         schema.json#/definitions/type"
1309     }
1310 },
1311 "required": [
1312     "href",
1313     "rt",
1314     "if"
1315 ],
1316 "type": "object"
1317 },
1318 "type": "array"
1319 }
1320 },
1321 "type": "object",
1322 "required": ["di", "links", "ttl"]
1323 }
1324 }
1325 }
1326

```

1327 **A.2.5 Property definition**

1328 Table A-2 defines the Properties that are part of the "oic.wk.rd" Resource Type.

1329 **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.

1330 **A.2.6 CRUDN behaviour**

1331 Table A-3 defines the CRUDN operations that are supported on the "oic.wk.rd" Resource Type.

1332 **Table A-3 – The CRUDN operations of the Resource with type "rt" = "oic.wk.rd".**

Create	Read	Update	Delete	Notify
	get	post		observe

1333 **A.3 CoAP Cloud Configuration Resource**

1334 **A.3.1 Introduction**

1335 The CoAPCloudConf Resource exposes configuration information for connecting to an OCF Cloud.

1336

1337 **A.3.2 Example URI**

1338 /CoAPCloudConfResURI

1339 **A.3.3 Resource type**

1340 The Resource Type is defined as: "oic.r.coapcloudconf".

1341 **A.3.4 OpenAPI 2.0 definition**

```

1342 {
1343   "swagger": "2.0",
1344   "info": {
1345     "title": "CoAP Cloud Configuration Resource",
1346     "version": "20190327",
1347     "license": {

```

```

1348     "name": "OCF Data Model License",
1349     "url":
1350 "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI
1351 CENSE.md",
1352     "x-copyright": "Copyright 2018-2019 Open Connectivity Foundation, Inc. All rights reserved."
1353 },
1354     "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
1355 },
1356     "schemes": ["http"],
1357     "consumes": ["application/json"],
1358     "produces": ["application/json"],
1359     "paths": {
1360     "/CoAPCloudConfResURI?if=oic.if.rw" : {
1361         "get": {
1362             "description": "The CoAPCloudConf Resource exposes configuration information for connecting
1363 to an OCF Cloud.\n",
1364             "parameters": [
1365                 {"$ref": "#/parameters/interface-all"}
1366             ],
1367             "responses": {
1368                 "200": {
1369                     "description": "",
1370                     "x-example":
1371                     {
1372                         "rt" : ["oic.r.coapcloudconf"],
1373                         "apn": "github",
1374                         "cis": "coaps+tcp://example.com:443",
1375                         "sid" : "987e6543-a21f-10d1-a112-421345746237",
1376                         "clec": 0
1377                     },
1378                     "schema": { "$ref": "#/definitions/CoAPCloudConf" }
1379                 }
1380             }
1381         },
1382         "post": {
1383             "description": "Update properties of the CoAPCloudConf Resource.\n",
1384             "parameters": [
1385                 {"$ref": "#/parameters/interface-all"},
1386                 {
1387                     "name": "body",
1388                     "in": "body",
1389                     "required": true,
1390                     "schema": { "$ref": "#/definitions/CoAPCloudConfUpdate" },
1391                     "x-example":
1392                     {
1393                         "at": "0f3d9f7fe5491d54077d",
1394                         "apn": "github",
1395                         "cis": "coaps+tcp://example.com:443",
1396                         "sid" : "987e6543-a21f-10d1-a112-421345746237"
1397                     }
1398                 }
1399             ],
1400             "responses": {
1401                 "200": {
1402                     "description": "",
1403                     "x-example":
1404                     {
1405                         "apn": "github",
1406                         "cis": "coaps+tcp://example.com:443",
1407                         "sid" : "987e6543-a21f-10d1-a112-421345746237",
1408                         "clec": 0
1409                     },
1410                     "schema": { "$ref": "#/definitions/CoAPCloudConf" }
1411                 }
1412             }
1413         },
1414     },
1415     "/CoAPCloudConfResURI?if=oic.if.baseline" : {
1416         "get": {
1417             "description": "The CoAPCloudConf Resource exposes configuration information for connecting
1418 to an OCF Cloud.\n",

```

```

1419     "parameters": [
1420       { "$ref": "#/parameters/interface-all" }
1421     ],
1422     "responses": {
1423       "200": {
1424         "description": "",
1425         "x-example":
1426           {
1427             "rt": ["oic.r.coapcloudconf"],
1428             "if": ["oic.if.rw", "oic.if.baseline"],
1429             "apn": "github",
1430             "cis": "coaps+tcp://example.com:443",
1431             "sid": "987e6543-a21f-10d1-a112-421345746237",
1432             "clec": 0
1433           },
1434         "schema": { "$ref": "#/definitions/CoAPCloudConf" }
1435       }
1436     },
1437   },
1438   "post": {
1439     "description": "Update Properties of the CoAPCloudConf Resource.\n",
1440     "parameters": [
1441       { "$ref": "#/parameters/interface-all" },
1442       {
1443         "name": "body",
1444         "in": "body",
1445         "required": true,
1446         "schema": { "$ref": "#/definitions/CoAPCloudConfUpdate" },
1447         "x-example":
1448           {
1449             "at": "0f3d9f7fe5491d54077d",
1450             "apn": "github",
1451             "cis": "coaps+tcp://example.com:443",
1452             "sid": "987e6543-a21f-10d1-a112-421345746237"
1453           }
1454       }
1455     ],
1456     "responses": {
1457       "200": {
1458         "description": "",
1459         "x-example":
1460           {
1461             "apn": "github",
1462             "cis": "coaps+tcp://example.com:443",
1463             "sid": "987e6543-a21f-10d1-a112-421345746237",
1464             "clec": 0
1465           },
1466         "schema": { "$ref": "#/definitions/CoAPCloudConf" }
1467       }
1468     }
1469   }
1470 },
1471 ],
1472 "parameters": {
1473   "interface-all": {
1474     "in": "query",
1475     "name": "if",
1476     "type": "string",
1477     "enum": ["oic.if.rw", "oic.if.baseline"]
1478   }
1479 },
1480 "definitions": {
1481   "CoAPCloudConf": {
1482     "properties": {
1483       "rt": {
1484         "description": "Resource Type of the Resource",
1485         "items": {
1486           "enum": ["oic.r.coapcloudconf"],
1487           "type": "string",
1488           "maxLength": 64
1489         }

```

```

1490         "minItems": 1,
1491         "uniqueItems": true,
1492         "readOnly": true,
1493         "type": "array"
1494     },
1495     "n" : {
1496         "$ref":
1497 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1498 schema.json#/definitions/n"
1499     },
1500     "cis" : {
1501         "description": "URL of OCF Cloud",
1502         "format": "uri",
1503         "type": "string"
1504     },
1505     "apn" : {
1506         "description": "The Authorisation Provider through which an Access Token was obtained.",
1507         "type": "string"
1508     },
1509     "sid" : {
1510         "$ref": "http://openconnectivityfoundation.github.io/core/schemas/oic.types-
1511 schema.json#/definitions/uuid"
1512     },
1513     "clec" : {
1514         "description": "Last Error Code during Cloud Provisioning (0: No Error, 1: Error response
1515 from the OCF Cloud, 2: Failed to connect to the OCF Cloud, 3: Failed to refresh Access Token, 4~254:
1516 Reserved, 255: Unknown error)",
1517         "enum": [
1518             0,
1519             1,
1520             2,
1521             3,
1522             255
1523         ],
1524         "readOnly": true
1525     },
1526     "id" : {
1527         "$ref":
1528 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1529 schema.json#/definitions/id"
1530     },
1531     "if" : {
1532         "description": "The OCF Interfaces supported by this Resource",
1533         "items": {
1534             "enum": [
1535                 "oic.if.rw",
1536                 "oic.if.baseline"
1537             ],
1538             "type": "string",
1539             "maxLength": 64
1540         },
1541         "minItems": 2,
1542         "uniqueItems": true,
1543         "readOnly": true,
1544         "type": "array"
1545     }
1546 },
1547 "type" : "object",
1548 "required":["cis", "sid"]
1549 },
1550 "CoAPCloudConfUpdate" : {
1551     "properties": {
1552         "cis" : {
1553             "description": "URL of OCF Cloud",
1554             "format": "uri",
1555             "type": "string"
1556         },
1557         "apn" : {
1558             "description": "The Authorisation Provider through which an Access Token was obtained.",
1559             "type": "string"
1560         },

```

```

1561         "at" : {
1562             "description": "Access Token which is returned by an Authorisation Provider or OCF
1563 Cloud.",
1564             "type": "string"
1565         },
1566         "sid" : {
1567             "$ref": "http://openconnectivityfoundation.github.io/core/schemas/oic.types-
1568 schema.json#/definitions/uuid"
1569         }
1570     },
1571     "type" : "object",
1572     "required":["cis", "at", "sid"]
1573 }
1574 }
1575 }
1576

```

1577 A.3.5 Property definition

1578 Table A.4 defines the Properties that are part of the "oic.r.coapcloudconf" Resource Type.

1579 **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.
clcc	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.

1580 **A.3.6 CRUDN behaviour**

1581 Table A.5 defines the CRUDN operations that are supported on the "oic.r.coapcloudconf" Resource
 1582 Type.

1583 **Table A.5 – The CRUDN operations of the Resource with type "rt" = "oic.r.coapcloudconf".**

Create	Read	Update	Delete	Notify
	get	post		observe

1584