

OCF Core Specification Extension CoAP Native Cloud

VERSION 2.0 | June 22, 2018



OPEN CONNECTIVITY
FOUNDATION®

CONTACT admin@openconnectivity.org
Copyright OCF © 2018. All Rights Reserved.

Legal Disclaimer

3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

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

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

Copyright © 2016-18 Open Connectivity Foundation, Inc. All rights reserved.

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

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

1	Scope	7
2	Normative references	7
3	Terms, definitions, symbols and abbreviations	8
3.1	Terms and definitions	8
3.2	Symbols and abbreviations	8
3.3	Conventions	8
3.4	Data types	8
4	Document conventions and organization	9
5	Overview	10
5.1	Introduction	10
5.2	Interaction Flow	10
5.3	Cloud Operational Flow	11
5.3.1	Pre-requisites and OCF Cloud User Account Creation	11
5.3.2	Mediator registration with the OCF Cloud	12
5.3.3	Device provisioning by the Mediator	12
5.3.4	Device Registration with the OCF Cloud	12
5.3.5	Connection with the OCF Cloud	12
5.3.6	Publishing Links to the OCF Cloud RD	13
5.3.7	Client to Server communication through the OCF Cloud	13
5.3.8	Refreshing connection with the OCF Cloud	13
5.3.9	Closing connection with the OCF Cloud	13
5.3.10	Deregistering from the OCF Cloud	13
5.4	Cloud Operational State Machine	14
6	Resource model	17
6.1	CoAPCloudConf Resource	17
6.1.1	Introduction	17
6.1.2	Resource Definition	17
6.1.3	Error Handling	18
7	Network and connectivity	18
8	Functional interactions	19
8.1	Onboarding, Provisioning, and Configuration	19
8.1.1	Overview	19
8.1.2	Use of Mediator	19
8.1.3	Device Connection to the OCF Cloud	22
8.1.4	Device Registration with the OCF Cloud	22
8.2	Resource Publication	22
8.3	Client Registration with the OCF Cloud	23
8.4	Resource Discovery	23
8.5	Device Deregistration from the OCF Cloud	25
9	Security	25

62	Annex A (normative) Resource Type definitions	26
63	A.1 List of Resource Type definitions	26
64	A.2 CoAP Cloud Configuration Resource	26
65	A.2.1 Introduction	26
66	A.2.2 Example URI	26
67	A.2.3 Resource Type	26
68	A.2.4 RAML Definition	26
69	A.2.5 Property Definition	29
70	A.2.6 CRUDN behaviour	30
71	Annex B (informative) Swagger2.0 definitions	31
72	B.1 CoAP Cloud Configuration Resource	31
73	B.1.1 Introduction	31
74	B.1.2 Example URI	31
75	B.1.3 Resource Type	31
76	B.1.4 Swagger2.0 Definition	31
77	B.1.5 Property Definition	36
78	B.1.6 CRUDN behaviour	37
79		
80		

81
82
83
84
85
86
87
88
89
90
91
92
93

Figures

Figure 1 OCF Cloud deployment architecture	10
Figure 2 Overall Operational State Machine	17
Figure 3 Registration with OCF Cloud.....	19
Figure 4 Device Provisioning by the Mediator	21
Figure 5 Resource publication to the OCF Cloud	23
Figure 6 Resource discovery through OCF Cloud	24
Figure 7 Request routing through OCF Cloud	25

Tables

94
95
96
97
98
99
100
101
102
103
104
105
106
107

Table 1 OCF Cloud Deployment Flow	11
Table 2 CoAPCloudConf Resource	17
Table 3 oic.r.coapcloudconf Resource Type definition	17
Table 4 Device - OCF Cloud Registration Flow	19
Table 5 Device Provisioning by the Mediator	21
Table 6. Alphabetized list of resources	26
Table 7 CoAP Cloud Configuration Resource Property Definitions	29
Table 8 CoAP Cloud Configuration Resource CRUDN operations	30
Table 9 The property definitions of the resource	36
Table 10 The CRUDN operations of the resource	37

108 **1 Scope**

109 This specification defines functional extensions to the capabilities defined in the OCF Core
110 Specification to meet the requirements of the OCF Cloud. This specification specifies new
111 Resource Types to enable the functionality and any extensions to the existing capabilities defined
112 in the OCF Core Specification.

113 **2 Normative references**

114 The following documents, in whole or in part, are normatively referenced in this document and are
115 indispensable for its application. For dated references, only the edition cited applies. For undated
116 references, the latest edition of the referenced document (including any amendments) applies.

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

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

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

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

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

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

123 OCF Core Specification Extension-Wi-Fi Easy Setup, *Open Connectivity Foundation Wi-Fi Easy
124 Setup Specification*, Version 1.3

125 Available at: [https://openconnectivity.org/specs/OCF_Core_Specification_Extension_Wi-
126 Fi_Easy_Setup_v1.3.0.pdf](https://openconnectivity.org/specs/OCF_Core_Specification_Extension_Wi-Fi_Easy_Setup_v1.3.0.pdf)

127 Latest version available at:

128 https://openconnectivity.org/specs/OCF_Core_Specification_Extension_Wi-Fi_Easy_Setup.pdf

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

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

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

136 IETF RFC 7159, *The JavaScript Object Notation (JSON) Data Interchange Format*, March 2014
137 <https://www.rfc-editor.org/info/rfc7159>

138 IETF RFC 7252, *The Constrained Application Protocol (CoAP)*, June 2014
139 <https://www.rfc-editor.org/info/rfc7252>

140 JSON Schema Validation, *JSON Schema: interactive and non-interactive validation*, January 2013
141 <http://json-schema.org/latest/json-schema-validation.html>

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

144

145 **3 Terms, definitions, symbols and abbreviations**

146 All terms and definitions as defined in the OCF Core Specification also apply to this specification.

147 **3.1 Terms and definitions**

148 As defined in the OCF Core Specification and OCF Security Specification with the following
149 additions

150 **3.1.1**

151 **Cloud Provider**

152 entity or organization that hosts an OCF Cloud.

153 **3.1.2**

154 **OCF Cloud**

155 an OCF Cloud is not an OCF Device, but a logical entity that is owned by the Cloud Provider. An
156 OCF Cloud is authorised to communicate with a Device on behalf of the OCF Cloud User.

157 **3.2 Symbols and abbreviations**

158 **3.2.1**

159 **UX**

160 User Experience

161 **3.3 Conventions**

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

166 **3.4 Data types**

167 As defined in the OCF Core Specification.

168

169 **4 Document conventions and organization**

170 In this document, features are described as required, recommended, allowed or DEPRECATED as
171 follows:

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

- 173 • These basic features shall be implemented to comply with Core Architecture. The phrases
174 “shall not”, and “PROHIBITED” indicate behaviour that is prohibited, i.e. that if performed
175 means the implementation is not in compliance.

176 Recommended (or should)(S).

- 177 • These features add functionality supported by Core Architecture and should be implemented.
178 Recommended features take advantage of the capabilities Core Architecture, usually without
179 imposing major increase of complexity. Notice that for compliance testing, if a recommended
180 feature is implemented, it shall meet the specified requirements to be in compliance with these
181 guidelines. Some recommended features could become requirements in the future. The phrase
182 “should not” indicates behaviour that is permitted but not recommended.

183 Allowed (may or allowed)(O).

- 184 • These features are neither required nor recommended by Core Architecture, but if the feature
185 is implemented, it shall meet the specified requirements to be in compliance with these
186 guidelines.

187 DEPRECATED.

- 188 • Although these features are still described in this specification, they should not be implemented
189 except for backward compatibility. The occurrence of a deprecated feature during operation of
190 an implementation compliant with the current specification has no effect on the
191 implementation’s operation and does not produce any error conditions. Backward compatibility
192 may require that a feature is implemented and functions as specified but it shall never be used
193 by implementations compliant with this specification.

194 Conditionally allowed (CA)

- 195 • The definition or behaviour depends on a condition. If the specified condition is met, then the
196 definition or behaviour is allowed, otherwise it is not allowed.

197 Conditionally required (CR)

- 198 • The definition or behaviour depends on a condition. If the specified condition is met, then the
199 definition or behaviour is required. Otherwise the definition or behaviour is allowed as default
200 unless specifically defined as not allowed.

201

202 Strings that are to be taken literally are enclosed in “double quotes”.

203 Words that are emphasized are printed in italic.

204

205 **5 Overview**

206 **5.1 Introduction**

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

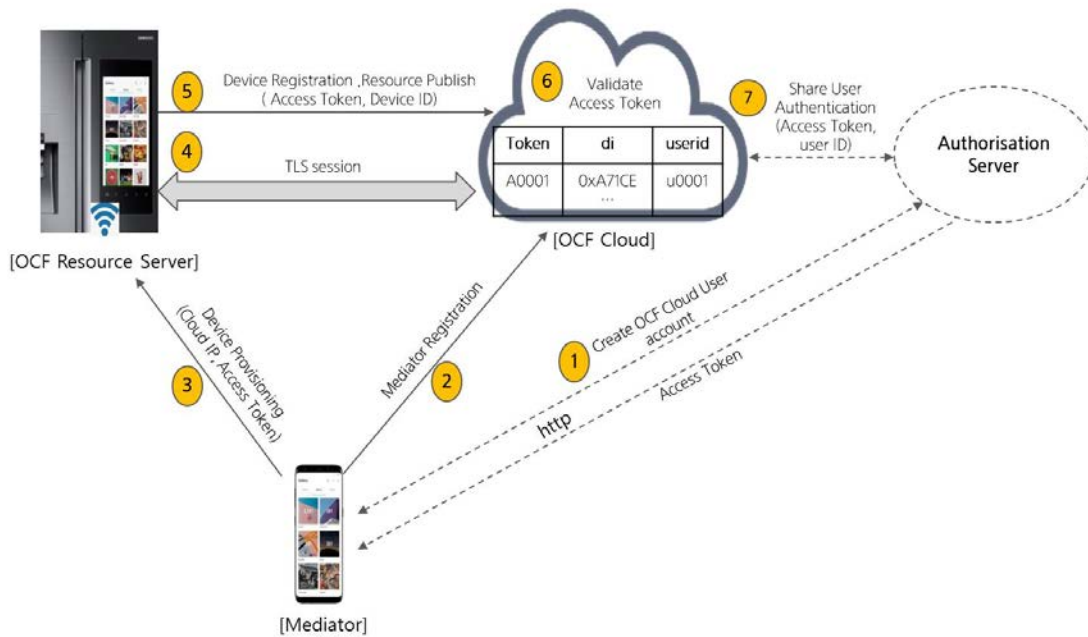
- 209 • CoAP over TCP protocol defined in OCF Core Specification
- 210 • Resource Directory defined in OCF Core Specification Section
- 211 • The requirements within this specification
- 212 • Security requirements and SVRs defined within the OCF Security Specification

213 Devices which are not within a single local network may interact with each other using CoAP over
 214 TCP (see OCF Core Specification) via an OCF Cloud. At any point in time, a Device is configured
 215 to use at most one OCF Cloud. The OCF Cloud groups Devices that belong to same OCF Cloud
 216 User under an OCF Cloud created User ID. All the Devices registered to the OCF Cloud and
 217 belonging to the same User ID can communicate with each other subject to the Device(s)
 218 authorising the OCF Cloud in the ACE2 policies.

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

221 **5.2 Interaction Flow**

222 This section describes how the elements with the overall OCF Cloud interact. Figure 1 provides
 223 an overall introduction:



OCF Cloud may act as the Authorisation Server to create the OCF Cloud User account.
 In that case, Step 1 may be done between the Mediator and OCF Cloud in which case #7 step is not required.

224

225

Figure 1 OCF Cloud deployment architecture

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.

226

Table 1 OCF Cloud Deployment Flow

227

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

- 230
- 231 • an account server function which is a logical entity that handles Device registration, Access Token validation and handles sign-in and token-refresh requests from the Device.
 - 232 • a Resource Directory as defined by the OCF Core Specification. The Resource Directory
233 exposes Resource information published by Devices. A Client, when discovering Devices,
234 receives a response from the Resource Directory on behalf of the Device. With information
235 included in the response from the Resource Directory, the Client may connect to the Device
236 via the OCF Cloud.

237 **5.3 Cloud Operational Flow**

238 The following sub-sections provide an informative overview of the flow which results on a Device
239 being registered with an OCF Cloud and Client interaction with that Device. The sections provide
240 references to the applicable Sections within this Specification and other Specifications that provide
241 normative details.

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

- 243 • Pre-requisites and OCF Cloud User account creation (Section 5.3.1)
- 244 • Mediator registration with the OCF Cloud (Section 5.3.2)
- 245 • Device provisioning by the Mediator (Section 5.3.3)
- 246 • Device registration with the OCF Cloud (Section 5.3.4)
- 247 • Device connection with the OCF Cloud (Section 5.3.5)
- 248 • Devices Publishing Links to the OCF Cloud RD (Section 5.3.6)
- 249 • Client to Server communication through the OCF Cloud (Section 5.3.7)
- 250 • Device refreshing connection with the OCF Cloud (Section 5.3.8)
- 251 • Device closing connection with the OCF Cloud (Section 5.3.9)
- 252 • Device de-registering from the OCF Cloud (Section 5.3.10)

253

254 **5.3.1 Pre-requisites and OCF Cloud User Account Creation**

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

257 The Device is onboarded to the OCF Network as defined in the OCF Security Specification.

258 The OCF Cloud User downloads a Mediator onto their personal device (e.g. phone) which will be
259 used to provision the Device. The Mediator is configured with or through some out of band

260 process to obtain the URL of the OCF Cloud (e.g. the Mediator may be an application from the
261 Cloud Provider).

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

264 **5.3.2 Mediator registration with the OCF Cloud**

265 See Sections 8.1.2.2, 0

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

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

273 **5.3.3 Device provisioning by the Mediator**

274 See Section 8.1.2.3; see also OCF Security Specification Section 7.5.1

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

281 **5.3.4 Device Registration with the OCF Cloud.**

282 See Sections 8.1.3, 8.1.4; see also OCF Security Specification Sections 10.4, 13.10

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

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

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

298 The Device is now registered with the OCF Cloud.

299 **5.3.5 Connection with the OCF Cloud**

300 See Section 8.1.4, see also OCF Security Specification Section 13.11

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

305 **5.3.6 Publishing Links to the OCF Cloud RD**

306 See Section 8.2; see also OCF Security Specification Section 10.4

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

310 **5.3.7 Client to Server communication through the OCF Cloud**

311 See Sections 0, 8.4; see also OCF Security Specification Section 10.4

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

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

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

318 **5.3.8 Refreshing connection with the OCF Cloud**

319 See OCF Security Specification Section 13.12

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

322 **5.3.9 Closing connection with the OCF Cloud**

323 See OCF Security Specification Section 13.11

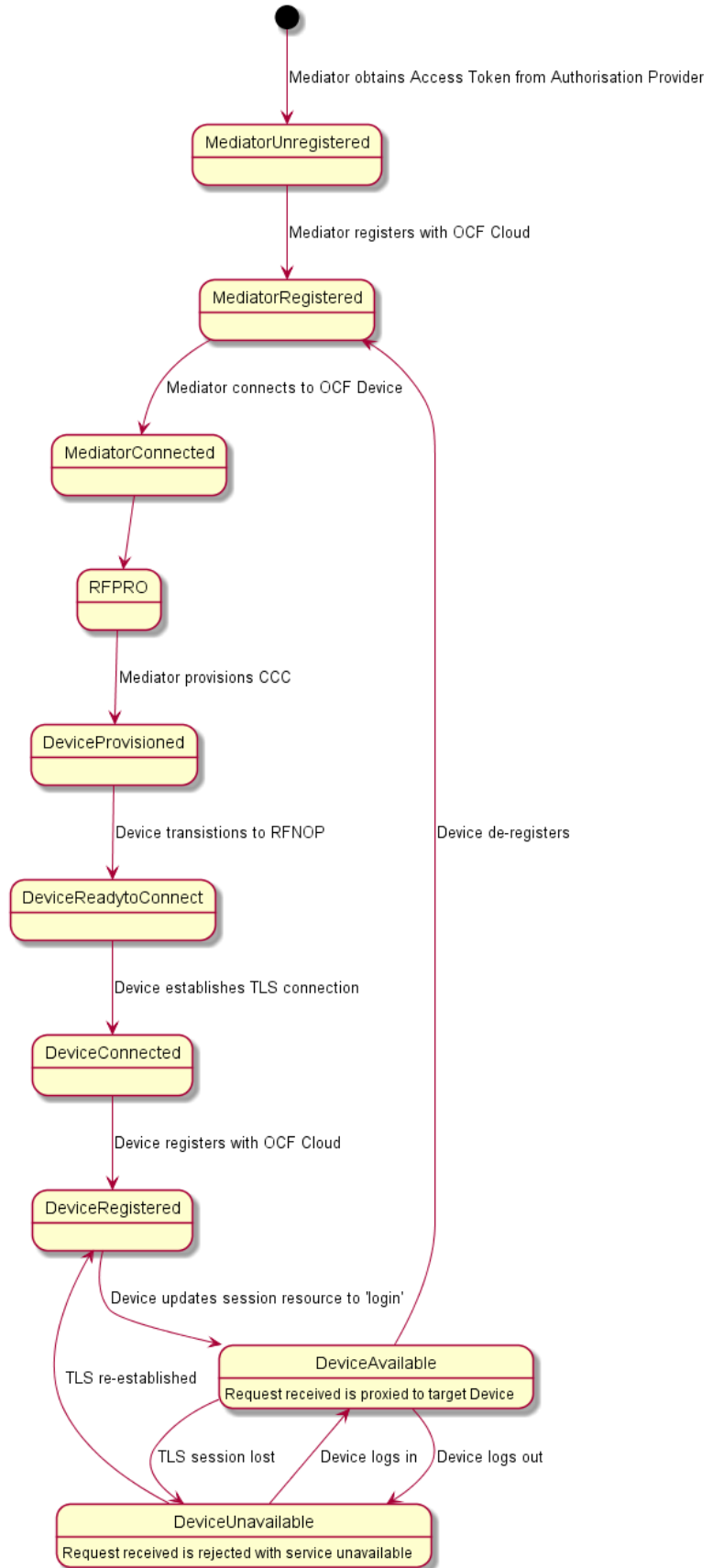
324 To log out of the OCF Cloud the Device sends an UPDATE request to the Session Resource
325 indicating a "login" status of "false". This does not delete or remove any of the Device
326 Registration information. The Device may log back into the OCF Cloud at any point prior to
327 expiration of the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

328 **5.3.10 Deregistering from the OCF Cloud**

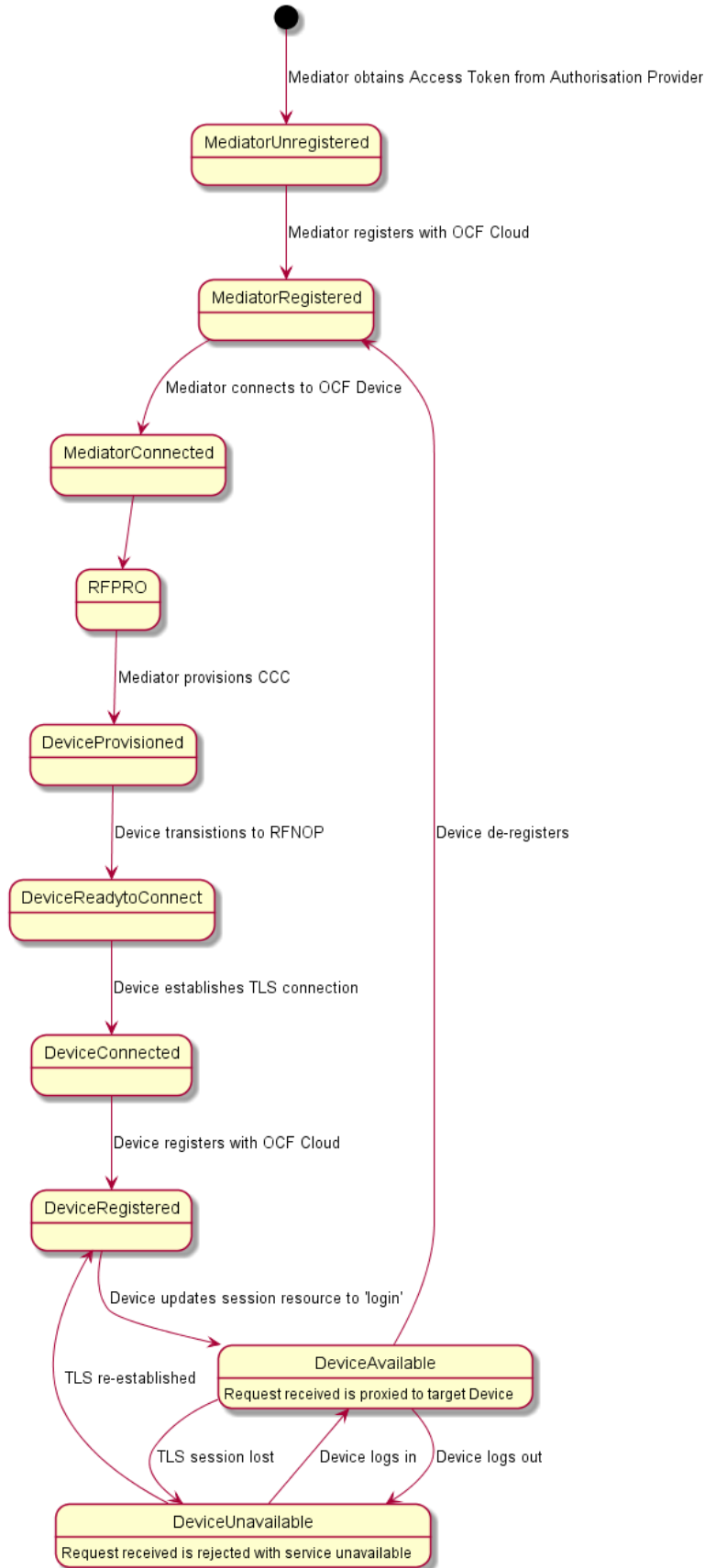
329 See Section 8.5; see also OCF Security Specification Section 13.10

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

333 To connect to the OCF Cloud again, the Device has to re-follow the flow starting with Mediator
334 provisioning (see Section 5.3.3).



336 Figure 2 Overall Operational State Machine captures the state machine that is described by the
337 informative operation flow provided in Section 5.3



339

Figure 2 Overall Operational State Machine

340 6 Resource model

341 6.1 CoAPCloudConf Resource

342 6.1.1 Introduction

343 The CoAPCloudConf resource exposes configuration information for connecting to an OCF Cloud.
 344 This is an optional discoverable Resource, which may additionally be included within the Easy
 345 Setup Collection (“oic.r.easyssetup”) and so used during the Easy Setup process as defined in OCF
 346 Core Specification Extension-Wi-Fi Easy Setup.

347

348 The CoAPCloudConf Resource shall expose only secure Endpoints (e.g. CoAPS); see the OCF
 349 Core Specification, Section 10.

350

351 6.1.2 Resource Definition

352 The CoAPCloudConf Resource is as defined in Table 2.

353

Table 2 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 3.	

354

355 Table 3 defines the details for the “oic.r.coapcloudconf” Resource Type.

356

Table 3 oic.r.coapcloudconf Resource Type definition

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Auth Provider Name	apn	String			RW	No	The name of the Authorisation Provider through which access token was obtained.
OCF Cloud interface URL	cis	String	uri		RW	Yes	URL of OCF Cloud.
Access Token	at	String	The Access Token is a string of at least one character		W ¹	Yes (in an UPDATE only)	Access token which is returned by an Authorisation Provider or OCF Cloud.
OCF Cloud UUID	sid	uuid			RW	Yes	The identity of the OCF Cloud
Last Error Code during	clcc	integer	enum		R	No	0: No Error, 1: Error response from the OCF Cloud,

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

Cloud Provisioning										2: Failed to connect to the OCF Cloud, 3: Failed to refresh Access Token, 4~254: Reserved, 255: Unknown error
---------------------------	--	--	--	--	--	--	--	--	--	--

357

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

359 **6.1.3 Error Handling**

360 The "clec" Property of the CoAPCloudConf Resource (i.e. “oic.r.coapcloudconf”) is used to indicate
361 any error that occurred in the cloud configuration process while trying to connect to the OCF Cloud
362 (using the information populated by the Mediator in the CoAPCloudConf Resource). This is an
363 optional Property and if implemented, is set by Device as defined below:

364 • The Device shall set the “clec” Property to 1 if it receives an error response from the OCF Cloud
365 (e.g. error response from the Cloud).

366 • The Device shall set the “clec” Property to 2 if there is a failure to connect to the OCF Cloud (e.g.
367 no reply, timeout, or timeout).

368 • The Device shall set the “clec” Property to 3 if it fails to refresh the Access Token (e.g. if it
369 receives an error response during the token refresh procedure).

370 **7 Network and connectivity**

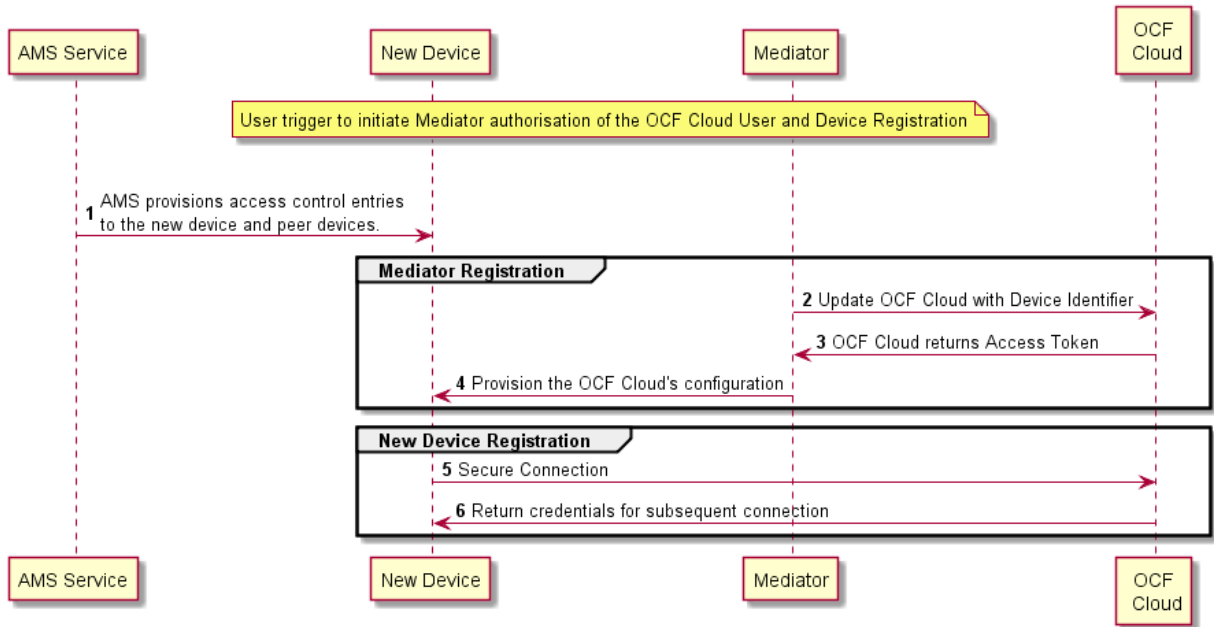
371 A TLS session exists between a Device and the OCF Cloud as specified in RFC 8323; this is
372 established following device configuration as detailed in Section 8.1.2.3.

373 **8 Functional interactions**

374 **8.1 Onboarding, Provisioning, and Configuration**

375 **8.1.1 Overview**

376 Figure 3 Registration with OCF Cloud below provides an overview of the interaction between the
 377 different entities to get the Device registered with the OCF Cloud. Details with respect to the flow
 378 are presented in the following Sections. A summary of the flow is provided in Table 4:



379
380
381
382

Figure 3 Registration with OCF Cloud

Steps	Description
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.

383 **Table 4 Device - OCF Cloud Registration Flow**

384 **8.1.2 Use of Mediator**

385 **8.1.2.1 Introduction**

386 The Mediator is a specialised service that is used for provisioning the “oic.r.coapcloudconf”
 387 Resource, and enabling connection of a headless Device to an OCF Cloud. The Mediator is
 388 specified in the OCF Core Specification Extension Wi-Fi Easy Setup.

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

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

- 395 • OCF Cloud Interface URL ("cis") Property
- 396 • OCF Cloud UUID ("sid") Property (to verify Cloud identity)
- 397 • Access Token ("at") Property that is validated by the OCF Cloud
- 398 • Optionally the Authorisation Provider name ("apn") Property through which the Access Token
399 was obtained

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

403 **8.1.2.2 OCF Cloud User Authorisation of the Mediator**

404 The Mediator uses a user authorisation mechanism to enable the OCF Cloud to validate the OCF
405 Cloud User's authorisation and obtain the OCF Cloud User's identity. The Authorisation Provider
406 should be trusted by both the OCF Cloud User and the OCF Cloud. The Mediator may use OAUTH
407 2.0 (see IETF RFC 6749) or another user authentication mechanism to obtain an Access Token as
408 a form of authorisation from an OCF Cloud User via an Authorisation Provider. This authorisation
409 achieves a variety of purposes. Firstly, the authorisation shows OCF Cloud User consent for
410 Mediator to connect to the OCF Cloud. Secondly, the authorisation is used to obtain information
411 to map the Devices to the same OCF Cloud User.

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

- 413 • Obtain an Access Token that is validated by the Cloud
- 414 • OCF Cloud User authorisation via an Authorisation Provider; this provides consent to connect
415 to the OCF Cloud.

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

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

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

423 The OCF Cloud then verifies the Access Token with the Authorisation Provider. If the Authorisation
424 Provider validates the Access Token successfully, then it will return information about the OCF
425 Cloud User to whom the Access Token belongs. The OCF Cloud generates a unique Access Token
426 for the Mediator (which may be the original Access Token from the Mediator or a new Access
427 Token) and a User ID (i.e. "uid" Property of "oic.r.account") if this is the first instance of registering
428 a Mediator with this OCF Cloud User. The User ID acts as a unique identity for the OCF Cloud
429 User. All instances of a Mediator for the same OCF Cloud User will be associated with the same
430 User ID. This information is returned to the Mediator over TLS. The returned Access Token and
431 User ID are used by the OCF Cloud to identify the Mediator. This returned Access Token is used
432 by the Mediator in subsequent interactions with the OCF Cloud.

433 All Devices registering with the OCF Cloud receive the same User ID from the OCF Cloud when
434 registering with the same Mediator.

435 **8.1.2.3 Device Provisioning by the Mediator**

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

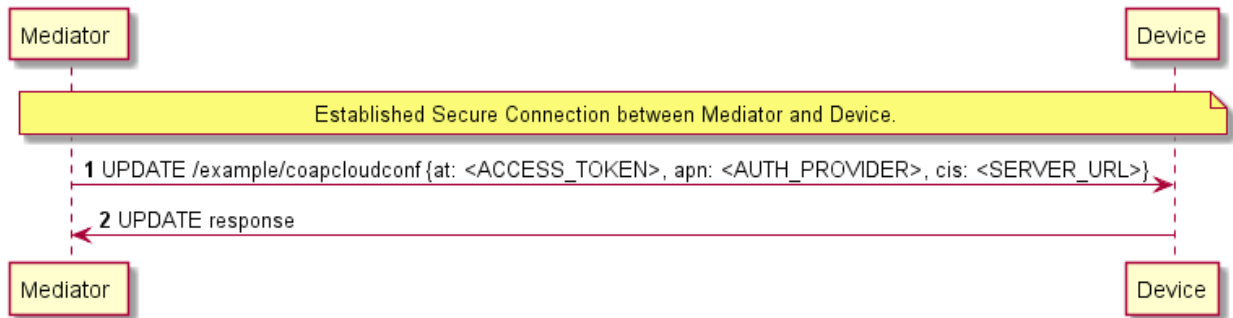
439 Once the OCF Cloud has associated the Mediator with a User ID, the Mediator can request the
 440 OCF Cloud to associate OCF Devices with the same User ID. To register the Device with the OCF
 441 Cloud, the Mediator first requests an Access Token for the Device from the OCF Cloud. The
 442 Mediator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

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

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

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

459 See OCF Security Specification Section 7.5.1 for details on the population of ACE2 entries on the
 460 Device to allow CRUDN operations from the Mediator and OCF Cloud.



461
 462 **Figure 4 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

463 **Table 5 Device Provisioning by the Mediator**

464 Please see OCF Security Specification Section 7.5.1 for further details on the mapping of
 465 Properties between the Device and OCF Cloud.

466 **8.1.3 Device Connection to the OCF Cloud**

467 On conclusion of Device provisioning as defined in Section 8.1.2.3 and after transitioning to a state
468 of RFNOP (if not already in RFNOP) the Device shall establish a TLS connection with the OCF
469 Cloud as defined in the OCF Security Specification Section 10.4. Further see the OCF Security
470 Specification Section 10.4.3 for additional security considerations.

471 If authentication of the TLS session being established as defined in the OCF Security Specification
472 fails, the "clec" Property of the "oic.r.coapcloudconf" Resource on the Device (if supported) shall
473 be updated about the failed state. If authentication succeeds, the Device and OCF Cloud establish
474 an encrypted link in accordance with the negotiated cipher suite. Further, if the TLS connection is
475 lost due to a failure the "clec" Property of the "oic.r.coapcloudconf" Resource on the Device (if
476 supported) should be updated about the failed state (value of "2").

477 If the TLS connection is lost either via a failure or closed by the OCF Cloud then it may be re-
478 established by following the procedures in the OCF Security Specification Section 10.4. A Device
479 may automatically attempt to re-establish the TLS connection, alternatively a Device may require
480 some user trigger to initiate the re-establishment of the TLS connection.

481 **8.1.4 Device Registration with the OCF Cloud**

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

486 After the TLS connection is established with the OCF Cloud, the Device shall register with the OCF
487 Cloud by sending an UPDATE request to "/oic/sec/account" as defined in Section 13.10 of the OCF
488 Security Specification. The OCF Cloud consequently associates the TLS connection with the
489 corresponding "uid" and "di" Properties populated in the "/oic/sec/account/" Resource. Any other
490 Device registering with the OCF Cloud is assigned the same User ID by the OCF Cloud when
491 registering with any Mediator associated with that User ID. Device Registration permits a Client to
492 access Resources on the OCF Cloud which are associated with the same User ID as the Client.

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

497 If the UPDATE operation is accepted by the OCF Cloud, the OCF Cloud responds as defined in
498 Section 13.10 of the OCF Security Specification.

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

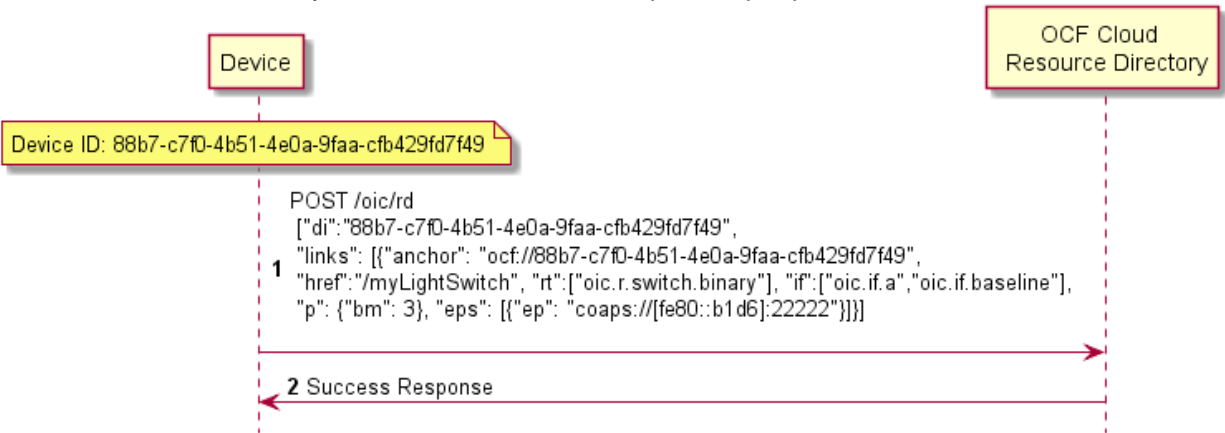
502 On completion of Device Registration the Device shall send an UPDATE to "/oic/sec/session" as
503 defined in Section 13.11 of the OCF Security Specification to ensure that the established TLS
504 session is maintained for subsequent interaction with the OCF Cloud Resource Directory as
505 defined in Section 8.2.

506 **8.2 Resource Publication**

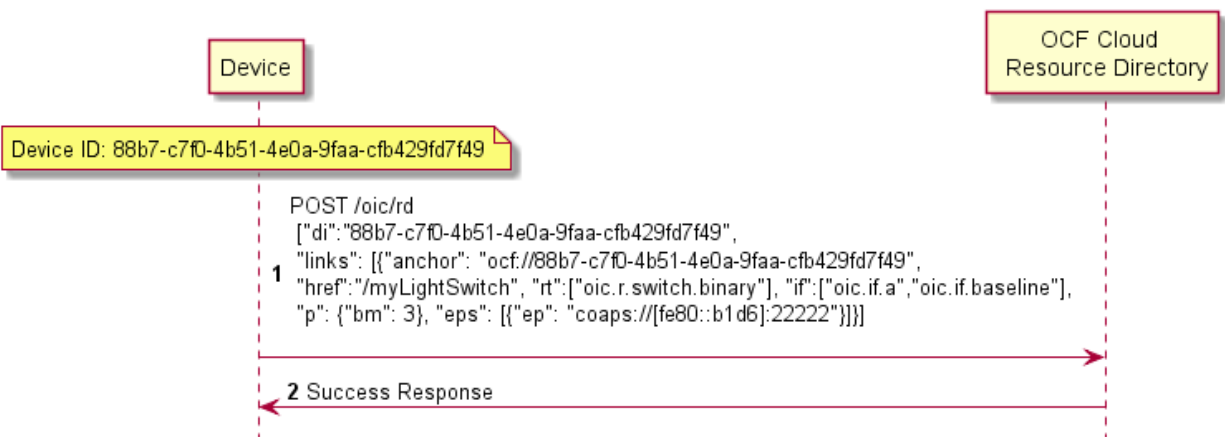
507 An OCF Cloud exposes a Resource Directory as defined in the OCF Core Specification Section
508 11.3.6. After a Device is registered with an OCF Cloud, the Device should publish its Resources
509 to the OCF Cloud's Resource Directory following the procedures defined in the OCF Core
510 Specification Section 11.3.6. The Device and OCF Cloud maintain a persistent TLS connection
511 over which requests received by the OCF Cloud for the Device are routed.

512 The OCF Cloud maintains an internal association between the published Endpoint information from
 513 the Device and the Endpoint information that it (the OCF Cloud) exposes in the Links within the
 514 OCF Cloud's Resource Directory. The Endpoint exposed by the OCF Cloud for all Resources
 515 published to it is that of the OCF Cloud itself and not the publishing Device. These Endpoints use
 516 a scheme of "coaps+tcp".

517 There is potential ambiguity where different instances of Devices from the same vendor (e.g. multiple
 518 lights) publish their Resources; this is because the local 'href' Link Parameter that is provided to the RD is
 519 likely to be the same in each case. In order to avoid this ambiguity the Resource Directory prepends the
 520 'href' that is published with the Device ID for the publishing Device. Thus ensuring that all requests
 521 received by the OCF Cloud have a unique URI per published Resource.



522
 523 Figure 5 Resource publication to the OCF Cloud for an example showing the provided Device ID
 524 from the Device; Figure 6 Resource discovery through OCF Cloud shows the pre-pending of the
 525 Device ID to the 'href' Link Parameter in the Resource Directory itself.



526
 527 **Figure 5 Resource publication to the OCF Cloud**

528 **8.3 Client Registration with the OCF Cloud**

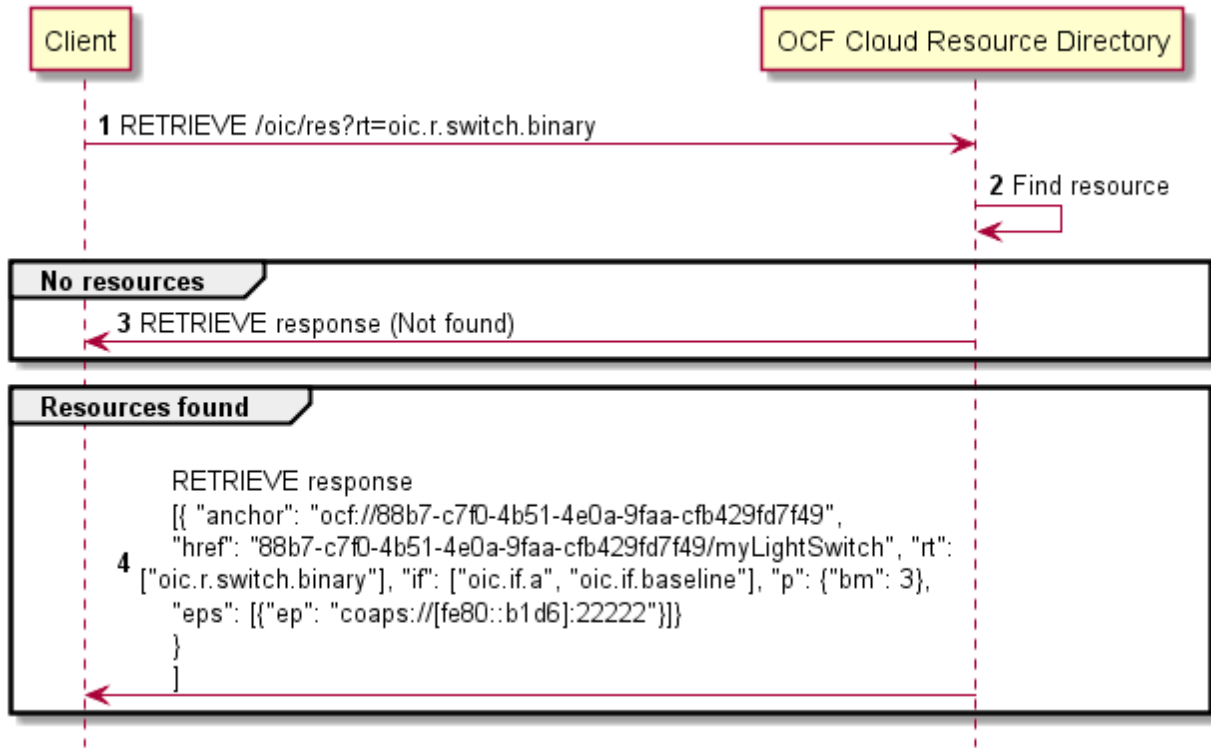
529 A Device acting in the Client role follows the same procedures as a Device in the Server role
 530 registering with the OCF Cloud. This Client is associated with a User ID in the same manner in
 531 which a Server is associated with the same User ID

532 **8.4 Resource Discovery**

533 A remote Device may query "/oic/res" to discover Resources published to the OCF Cloud. The
 534 OCF Cloud's Resource Directory responds with Links for the Resources published to the OCF

535 Cloud by Devices that are registered to the OCF Cloud for the User ID with which the remote
 536 Device is associated. The “eps” Link Parameter in the “/oic/res” response are for the OCF Cloud
 537 and not the publishing Device.

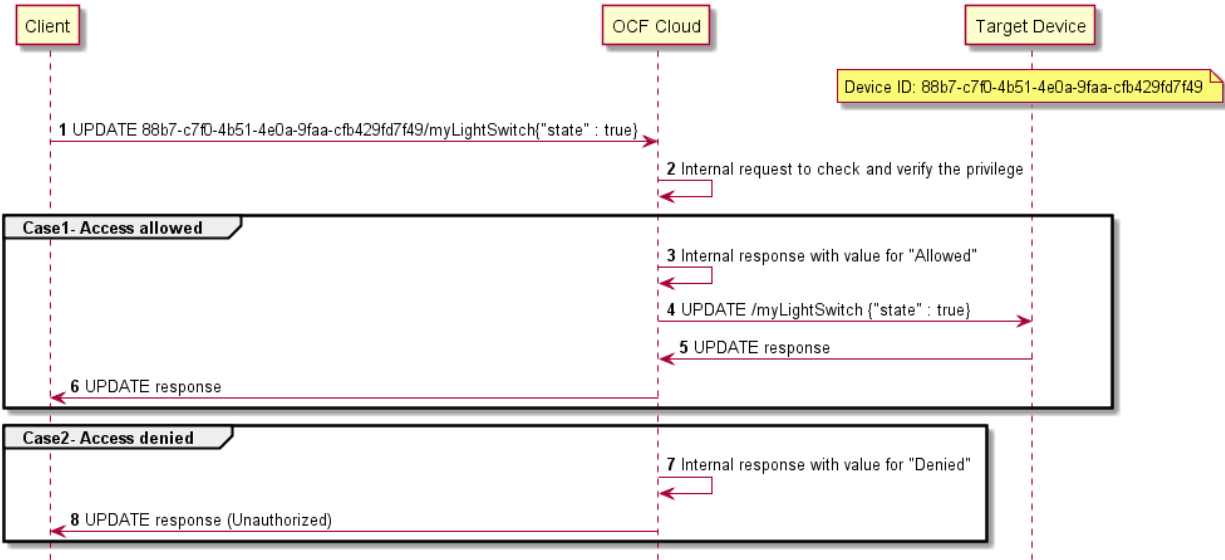
538 See Figure 6 Resource discovery through OCF Cloud for an illustrative flow for Resource Discovery,
 539 note the population of the ‘href’ for instance of “oic.r.switch.binary” including the Device ID of the
 540 target Device in accordance with section 8.2:



541
 542 **Figure 6 Resource discovery through OCF Cloud**

543 The OCF Cloud acts as a simple proxy, forwarding the messages to the publishing Devices. The
 544 remote Device sends a RETRIEVE to the OCF Cloud to obtain the content of the Server’s published
 545 Resources, the OCF Cloud will route the message to the target Device after first removing the
 546 Device ID that had been prepended to the ‘href’ Link Parameter by the Cloud RD. Similarly, other
 547 CRUDN operations originated by a Client are routed to the Server via the OCF Cloud. The
 548 publishing Device treats the forwarded request message as a request from the OCF Cloud. The
 549 publishing Device authorises the request as specified in the OCF Security Specification, using the
 550 UUID of the OCF Cloud configured in the “sid” Property of “oic.r.coapcloudconf”. The publishing
 551 Device sends a response message to the OCF Cloud, and the OCF Cloud forwards the response
 552 to the Client which sent the corresponding request.

553 Figure 7 Request routing through OCF Cloud illustrates request routing via the OCF Cloud



554
555

556

Figure 7 Request routing through OCF Cloud

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

559 **8.5 Device Deregistration from the OCF Cloud**

560 To deregister from the OCF Cloud the Device first sends a DELETE operation to the
561 "/oic/sec/account" Resource as defined in the OCF Security Specification Section 13.10.

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

564 **9 Security**

565 OCF Cloud security requirements are captured in the OCF Security Specification.

Annex A (normative)

Resource Type definitions

A.1 List of Resource Type definitions

Table 6 contains the list of defined resources in this specification.

Table 6. Alphabetized list of resources

Friendly Name (informative)	Resource Type (rt)	Section
CoAP Configuration	Cloud "oic.r.coapcloudconf"	A.2

A.2 CoAP Cloud Configuration Resource

A.2.1 Introduction

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

A.2.2 Example URI

/CoAPCloudConfResURI

A.2.3 Resource Type

The resource type (rt) is defined as: oic.r.coapcloudconf.

A.2.4 RAML Definition

```
581 #%RAML 0.8
582 title: CoAP Cloud Configuration Resource
583 version: v0.0.3-20180116
584 traits:
585   - interface-rw :
586     queryParameters:
587       if:
588         enum: ["oic.if.rw"]
589   - interface-baseline :
590     queryParameters:
591       if:
592         enum: ["oic.if.baseline"]
593   - interface-all :
594     queryParameters:
595       if:
596         enum: ["oic.if.baseline", "oic.if.rw"]
597
598 /CoAPCloudConfResURI?if=oic.if.baseline:
599   description: |
600     The CoAPCloudConf Resource exposes configuration information for connecting to an OCF Cloud.
601
602   is : ['interface-baseline']
603   get:
604     description: |
```

```

605
606     responses :
607         200:
608             body:
609                 application/json:
610                     schema: /
611                         {
612                             "$schema": "http://json-schema.org/draft-04/schema#",
613                             "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights
614 reserved.",
615                             "id": "http://openconnectivityfoundation.github.io/core-
616 extensions/schemas/oic.r.coapcloudconf-schema.json#",
617                             "definitions": {
618                                 "oic.r.coapcloudconf": {
619                                     "type": "object",
620                                     "properties": {
621                                         "apn": {
622                                             "type": "string",
623                                             "description": "The Authorisation Provider through which an Access Token
624 was obtained."
625                                         },
626                                         "cis": {
627                                             "type": "string",
628                                             "description": "URL of OCF Cloud",
629                                             "format": "uri"
630                                         },
631                                         "sid": {
632                                             "$ref":
633 "http://openconnectivityfoundation.github.io/core/schemas/oic.types-schema.json#/definitions/uuid",
634                                             "description": "The identity of the OCF Cloud"
635                                         },
636                                         "clec": {
637                                             "enum": [0, 1, 2, 3, 255],
638                                             "description": "Last Error Code during Cloud Provisioning (0: No Error, 1:
639 Error response from the OCF Cloud, 2: Failed to connect to the OCF Cloud, 3: Failed to refresh
640 Access Token, 4~254: Reserved, 255: Unknown error)",
641                                             "readOnly": true
642                                         }
643                                     },
644                                     "required":["cis", "sid"]
645                                 }
646                             },
647                             "type": "object",
648                             "allOf": [
649                                 { "$ref": "http://openconnectivityfoundation.github.io/core/schemas/oic.core-
650 schema.json#/definitions/oic.core"},
651                                 { "$ref": "#/definitions/oic.r.coapcloudconf" }
652                             ]
653                         }
654
655                     example: /
656                         {
657                             "rt": ["oic.r.coapcloudconf"],
658                             "if" : ["oic.if.baseline", "oic.if.rw"],
659                             "apn": "github",
660                             "cis": "coaps+tcp://example.com:443",
661                             "sid" : "987e6543-a21f-10d1-a112-421345746237",
662                             "clec": 0
663                         }
664
665                 post:
666                     description: |
667                         Update properties of CoAPCloudConf resource.
668
669                 body:

```

```

670     application/json:
671         schema: /
672             {
673                 "$schema": "http://json-schema.org/draft-04/schema#",
674                 "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights
675 reserved.",
676                 "id": "http://openconnectivityfoundation.github.io/core-
677 extensions/schemas/oic.r.coapcloudconf-update-schema.json#",
678                 "definitions": {
679                     "oic.r.coapcloudconf": {
680                         "type": "object",
681                         "properties": {
682                             "apn": {
683                                 "type": "string",
684                                 "description": "The Authorisation Provider through which an Access Token was
685 obtained."
686                             },
687                             "cis": {
688                                 "type": "string",
689                                 "description": "URL of OCF Cloud",
690                                 "format": "uri"
691                             },
692                             "at": {
693                                 "type": "string",
694                                 "description": "Access Token which is returned by an Authorisation Provider or
695 OCF Cloud.",
696                                 "pattern": "(?!$|\\s+).*"
697                             },
698                             "sid": {
699                                 "$ref": "http://openconnectivityfoundation.github.io/core/schemas/oic.types-
700 schema.json#/definitions/uuid",
701                                 "description": "The identity of the OCF Cloud"
702                             }
703                         },
704                         "required":["cis", "at", "sid"]
705                     }
706                 },
707                 "type": "object",
708                 "allOf": [
709                     { "$ref": "http://openconnectivityfoundation.github.io/core/schemas/oic.core-
710 schema.json#/definitions/oic.core"},
711                     { "$ref": "#/definitions/oic.r.coapcloudconf" }
712                 ]
713             }
714
715         example: /
716             {
717                 "at": "0f3d9f7fe5491d54077d",
718                 "apn": "github",
719                 "cis": "coaps+tcp://example.com:443",
720                 "sid" : "987e6543-a21f-10d1-a112-421345746237"
721             }
722
723     responses :
724         200:
725             body:
726                 application/json:
727                     schema: /
728                         {
729                             "$schema": "http://json-schema.org/draft-04/schema#",
730                             "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights
731 reserved.",
732                             "id": "http://openconnectivityfoundation.github.io/core-
733 extensions/schemas/oic.r.coapcloudconf-schema.json#",
734                             "definitions": {
735                                 "oic.r.coapcloudconf": {

```

```

736         "type": "object",
737         "properties": {
738             "apn": {
739                 "type": "string",
740                 "description": "The Authorisation Provider through which an Access Token
741 was obtained."
742             },
743             "cis": {
744                 "type": "string",
745                 "description": "URL of OCF Cloud",
746                 "format": "uri"
747             },
748             "sid": {
749                 "$ref":
750 "http://openconnectivityfoundation.github.io/core/schemas/oic.types-schema.json#/definitions/uuid",
751                 "description": "The identity of the OCF Cloud"
752             },
753             "clec": {
754                 "enum": [0, 1, 2, 3, 255],
755                 "description": "Last Error Code during Cloud Provisioning (0: No Error, 1:
756 Error response from the OCF Cloud, 2: Failed to connect to the OCF Cloud, 3: Failed to refresh
757 Access Token, 4~254: Reserved, 255: Unknown error)",
758                 "readOnly": true
759             }
760         },
761         "required":["cis", "sid"]
762     }
763 },
764 "type": "object",
765 "allOf": [
766     { "$ref": "http://openconnectivityfoundation.github.io/core/schemas/oic.core-
767 schema.json#/definitions/oic.core" },
768     { "$ref": "#/definitions/oic.r.coapcloudconf" }
769 ]
770 }
771
772 example: /
773 {
774     "apn": "github",
775     "cis": "coaps+tcp://example.com:443",
776     "sid" : "987e6543-a21f-10d1-a112-421345746237",
777     "clec": 0
778 }
779

```

780 A.2.5 Property Definition

781 **Table 7 CoAP Cloud Configuration Resource Property Definitions**

Property name	Value type	Mandatory	Access mode	Description
cis	string	yes		URL of OCF Cloud
clec	multiple types: see schema		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)
apn	string			The Authorisation Provider through which an Access Token was obtained.
sid	multiple types: see schema	yes		The identity of the OCF Cloud
at	string	yes (UPDATE only)	Write Only	Access Token which is returned by an Authorisation Provider or OCF Cloud.

782 **A.2.6 CRUDN behaviour**

783 **Table 8 CoAP Cloud Configuration Resource CRUDN operations**

Resource	Create	Read	Update	Delete	Notify
/CoAPCloudConfResURI		get	post		

784

785 **Annex B (informative)**
786
787 **Swagger2.0 definitions**

788 **B.1 CoAP Cloud Configuration Resource**

789 **B.1.1 Introduction**

790 The CoAPCloudConf Resource exposes configuration information for connecting to an OCF
791 Cloud.
792

793 **B.1.2 Example URI**

794 /CoAPCloudConfResURI

795 **B.1.3 Resource Type**

796 The resource type (rt) is defined as: ['oic.r.coapcloudconf'].

797 **B.1.4 Swagger2.0 Definition**

```
798 {  
799   "swagger": "2.0",  
800   "info": {  
801     "title": "CoAP Cloud Configuration Resource Read Write Interface",  
802     "version": "v0.0.3-20180116",  
803     "license": {  
804       "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",  
805       "x-description": "Redistribution and use in source and binary forms, with or without  
806 modification, are permitted provided that the following conditions are met:\n      1.  
807 Redistributions of source code must retain the above copyright notice, this list of conditions and  
808 the following disclaimer.\n      2. Redistributions in binary form must reproduce the above  
809 copyright notice, this list of conditions and the following disclaimer in the documentation and/or  
810 other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open  
811 Connectivity Foundation, INC. \AS IS\ AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT  
812 LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR  
813 WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n\n      IN NO EVENT SHALL THE Open Connectivity  
814 Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,  
815 EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS  
816 OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n\n      HOWEVER CAUSED AND  
817 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR  
818 OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY  
819 OF SUCH DAMAGE.\n"      }  
820   },  
821   },  
822   "schemes": ["http"],  
823   "consumes": ["application/json"],  
824   "produces": ["application/json"],  
825   "paths": {  
826     "/CoAPCloudConfResURI?if=oic.if.rw" : {  
827       "get": {  
828         "description": "The CoAPCloudConf Resource exposes configuration information for connecting  
829 to an OCF Cloud.\nRetrieve properties of CoAPCloudConf resource.\n",  
830         "parameters": [  
831           { "$ref": "#/parameters/interface-rw" }  
832         ],  
833         "responses": {  
834           "200": {  
835             "description": "",  
836             "x-example": {  
837               "apn": "github",  
838               "cis": "coaps+tcp://example.com:443",  
839               "sid": "987e6543-a21f-10d1-a112-421345746237",  
840               "clec": 0  
841             }  
842           }  
843         },  
844         "schema": { "$ref": "#/definitions/CoAPCloudConf" }  
845       }  
846     }  
847   }  
848 }
```

```

846     }
847   },
848   "post": {
849     "description": "Update properties of CoAPCloudConf resource.\n",
850     "parameters": [
851       { "$ref": "#/parameters/interface-rw" },
852       {
853         "name": "body",
854         "in": "body",
855         "required": true,
856         "schema": { "$ref": "#/definitions/CoAPCloudConfUpdate" },
857         "x-example":
858           {
859             "at": "0f3d9f7fe5491d54077d",
860             "apn": "github",
861             "cis": "coaps+tcp://example.com:443",
862             "sid": "987e6543-a21f-10d1-a112-421345746237"
863           }
864       }
865     ],
866     "responses": {
867       "200": {
868         "description": "",
869         "x-example":
870           {
871             "apn": "github",
872             "cis": "coaps+tcp://example.com:443",
873             "sid": "987e6543-a21f-10d1-a112-421345746237",
874             "clec": 0
875           }
876         ,
877         "schema": { "$ref": "#/definitions/CoAPCloudConf" }
878       }
879     }
880   },
881 },
882 "/CoAPCloudConfResURI?if=oic.if.baseline" : {
883   "get": {
884     "description": "The CoAPCloudConf Resource exposes configuration information for connecting
885 to an OCF Cloud.\n",
886     "parameters": [
887       { "$ref": "#/parameters/interface-baseline" }
888     ],
889     "responses": {
890       "200": {
891         "description": "",
892         "x-example":
893           {
894             "rt": ["oic.r.coapcloudconf"],
895             "if": ["oic.if.baseline", "oic.if.rw"],
896             "apn": "github",
897             "cis": "coaps+tcp://example.com:443",
898             "sid": "987e6543-a21f-10d1-a112-421345746237",
899             "clec": 0
900           }
901         ,
902         "schema": { "$ref": "#/definitions/CoAPCloudConf" }
903       }
904     }
905   },
906   "post": {
907     "description": "Update properties of CoAPCloudConf resource.\n",
908     "parameters": [
909       { "$ref": "#/parameters/interface-baseline" },
910       {
911         "name": "body",
912         "in": "body",
913         "required": true,
914         "schema": { "$ref": "#/definitions/CoAPCloudConfUpdate" },
915         "x-example":
916           {

```



```

917         "at": "0f3d9f7fe5491d54077d",
918         "apn": "github",
919         "cis": "coaps+tcp://example.com:443",
920         "sid": "987e6543-a21f-10d1-a112-421345746237"
921     }
922 },
923 ],
924 "responses": {
925     "200": {
926         "description": "",
927         "x-example":
928         {
929             "apn": "github",
930             "cis": "coaps+tcp://example.com:443",
931             "sid": "987e6543-a21f-10d1-a112-421345746237",
932             "clec": 0
933         }
934     },
935     "schema": { "$ref": "#/definitions/CoAPCloudConf" }
936 }
937 },
938 },
939 },
940 },
941 "parameters": {
942     "interface-rw" : {
943         "in" : "query",
944         "name" : "if",
945         "type" : "string",
946         "enum" : ["oic.if.rw"]
947     },
948     "interface-baseline" : {
949         "in" : "query",
950         "name" : "if",
951         "type" : "string",
952         "enum" : ["oic.if.baseline"]
953     },
954     "interface-all" : {
955         "in" : "query",
956         "name" : "if",
957         "type" : "string",
958         "enum" : ["oic.if.baseline", "oic.if.rw"]
959     }
960 },
961 "definitions": {
962     "CoAPCloudConf" : {
963         "properties": {
964             "rt" :
965             {
966                 "description": "Resource Type of the Resource",
967                 "items": {
968                     "maxLength": 64,
969                     "type": "string"
970                 },
971                 "minItems": 1,
972                 "readOnly": true,
973                 "type": "array"
974             },
975             "n" :
976             {
977                 "description": "Friendly name of the resource",
978                 "maxLength": 64,
979                 "readOnly": true,
980                 "type": "string"
981             },
982             "cis" :
983             {
984                 "description": "URL of OCF Cloud",
985                 "format": "uri",
986             }
987         }

```

```

988         "type": "string"
989     },
990
991     "apn" :
992     {
993         "description": "The Authorisation Provider through which an Access Token was obtained.",
994         "type": "string"
995     },
996
997     "sid" :
998     {
999         "description": "Format pattern according to IETF RFC 4122.",
1000        "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
1001 9]{12}$",
1002        "type": "string"
1003    },
1004
1005    "clec" :
1006    {
1007        "description": "Last Error Code during Cloud Provisioning (0: No Error, 1: Error response
1008 from the OCF Cloud, 2: Failed to connect to the OCF Cloud, 3: Failed to refresh Access Token,
1009 4~254: Reserved, 255: Unknown error)",
1010        "enum": [
1011            0,
1012            1,
1013            2,
1014            3,
1015            255
1016        ],
1017        "readOnly": true
1018    },
1019
1020    "id" :
1021    {
1022        "description": "Instance ID of this specific resource",
1023        "maxLength": 64,
1024        "readOnly": true,
1025        "type": "string"
1026    },
1027
1028    "if" :
1029    {
1030        "description": "The interface set supported by this resource",
1031        "items": {
1032            "enum": [
1033                "oic.if.baseline",
1034                "oic.if.ll",
1035                "oic.if.b",
1036                "oic.if.lb",
1037                "oic.if.rw",
1038                "oic.if.r",
1039                "oic.if.a",
1040                "oic.if.s"
1041            ],
1042            "type": "string"
1043        },
1044        "minItems": 1,
1045        "readOnly": true,
1046        "type": "array"
1047    }
1048 }
1049 }
1050 , "type" : "object"
1051 }
1052 ,
1053 "CoAPCloudConfUpdate" : {
1054     "properties": {
1055         "rt" :
1056         {
1057             "description": "Resource Type of the Resource",
1058             "items": {

```

```

1059         "maxLength": 64,
1060         "type": "string"
1061     },
1062     "minItems": 1,
1063     "readOnly": true,
1064     "type": "array"
1065 },
1066
1067 "n" :
1068     {
1069         "description": "Friendly name of the resource",
1070         "maxLength": 64,
1071         "readOnly": true,
1072         "type": "string"
1073     },
1074
1075 "cis" :
1076     {
1077         "description": "URL of OCF Cloud",
1078         "format": "uri",
1079         "type": "string"
1080     },
1081
1082 "apn" :
1083     {
1084         "description": "The Authorisation Provider through which an Access Token was obtained.",
1085         "type": "string"
1086     },
1087
1088 "at" :
1089     {
1090         "description": "Access Token which is returned by an Authorisation Provider or OCF
1091 Cloud.",
1092         "type": "string.",
1093         "pattern": "(?!$|\\s+).*"
1094     },
1095
1096 "sid" :
1097     {
1098         "description": "Format pattern according to IETF RFC 4122.",
1099         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
1100 9]{12}$",
1101         "type": "string"
1102     },
1103
1104 "id" :
1105     {
1106         "description": "Instance ID of this specific resource",
1107         "maxLength": 64,
1108         "readOnly": true,
1109         "type": "string"
1110     },
1111
1112 "if" :
1113     {
1114         "description": "The interface set supported by this resource",
1115         "items": {
1116             "enum": [
1117                 "oic.if.baseline",
1118                 "oic.if.ll",
1119                 "oic.if.b",
1120                 "oic.if.lb",
1121                 "oic.if.rw",
1122                 "oic.if.r",
1123                 "oic.if.a",
1124                 "oic.if.s"
1125             ],
1126             "type": "string"
1127         },
1128         "minItems": 1,
1129         "readOnly": true,

```

```

1130         "type": "array"
1131     }
1132
1133     }
1134     , "type" : "object"
1135 }
1136 }
1137 }
1138

```

1139 B.1.5 Property Definition

1140 **Table 9 The property definitions of the resource**

Property name	Value type	Mandatory	Access mode	Description
rt	array: see schema		Read Only	Resource Type of the Resource
cis	string	yes	Read Write	URL of OCF Cloud
sid	string	yes	Read Write	Format pattern according to IETF RFC 4122.
apn	string		Read Write	The Authorisation Provider through which an Access Token was obtained.
clec	multiple types: see schema		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)
id	string		Read Only	Instance ID of this specific resource
n	string		Read Only	Friendly name of the resource
if	array: see schema		Read Only	The interface set supported by this resource

rt	array: schema	see		Read Only	Resource Type of the Resource
cis	string	yes		Read Write	URL of OCF Cloud
sid	string	yes		Read Write	Format pattern according to IETF RFC 4122.
apn	string			Read Write	The Authorisation Provider through which an Access Token was obtained.
at	string	yes only	(UPDATE)	Write Only	Access Token which is returned by an Authorisation Provider or OCF Cloud.
id	string			Read Only	Instance ID of this specific resource
n	string			Read Only	Friendly name of the resource
if	array: schema	see		Read Only	The interface set supported by this resource

1141 **B.1.6 CRUDN behaviour**

1142 **Table 10 The CRUDN operations of the resource**

Resource	Create	Read	Update	Delete	Notify
/CoAPCloudConfResURI		get	post		observe

1143
1144
1145