

OCF Cloud Security Specification

VERSION 2.2.0 | July 2020



OPEN CONNECTIVITY
FOUNDATION™

CONTACT admin@openconnectivity.org

Copyright Open Connectivity Foundation, Inc. © 2020.
All Rights Reserved.

1 **LEGAL DISCLAIMER**

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

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

15 Copyright © 2019-2020 Open Connectivity Foundation, Inc. All rights reserved.

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

17 **TABLE OF CONTENTS**

18 1 Purpose and Role 1

19 2 Normative References 1

20 3 Terms, definitions, and abbreviated terms 1

21 3.1 Terms and definitions..... 1

22 3.2 Abbreviated terms..... 2

23 4 Document Conventions and Organization 3

24 4.1 Conventions..... 3

25 4.2 Notation..... 3

26 4.3 Data types 4

27 4.4 Document structure..... 4

28 5 Security overview 5

29 5.1 Preamble 5

30 5.2 Device Provisioning for OCF Cloud and Device Registration Overview..... 5

31 5.3 Credential overview 5

32 6 Device provisioning for OCF Cloud..... 5

33 6.1 Cloud Provisioning General 5

34 6.2 Device Provisioning by Mediator 6

35 7 Device authentication with OCF Cloud..... 8

36 7.1 Device Authentication with OCF Cloud General 8

37 7.2 Device Connection with the OCF Cloud 8

38 7.3 Security Considerations 10

39 8 Message integrity and confidentiality 10

40 8.1 Cloud Session Semantics 10

41 8.2 Cipher suites for OCF Cloud Credentials 10

42 9 Security resources..... 11

43 9.1 Account Resource..... 11

44 9.2 Account Session resource..... 12

45 9.3 Account Token Refresh Resource 13

46 10 Security hardening guidelines..... 14

47 10.1 Security hardening guidelines general 14

48 Annex A (normative) Resource Type definitions 16

49 A.1 Account Token..... 16

50 A.1.1 Introduction 16

51 A.1.2 Well-known URI..... 16

52 A.1.3 Resource type 16

53 A.1.4 OpenAPI 2.0 definition..... 16

54 A.1.5 Property definition 19

55 A.1.6 CRUDN behaviour 20

56 A.2 Session..... 20

57 A.2.1 Introduction 20

58 A.2.2 Well-known URI..... 20

59	A.2.3	Resource type	20
60	A.2.4	OpenAPI 2.0 definition.....	20
61	A.2.5	Property definition	22
62	A.2.6	CRUDN behaviour	23
63	A.3	Token Refresh	23
64	A.3.1	Introduction	23
65	A.3.2	Well-known URI	24
66	A.3.3	Resource type	24
67	A.3.4	OpenAPI 2.0 definition.....	24
68	A.3.5	Property definition	26
69	A.3.6	CRUDN behaviour	27
70			

71	FIGURES	
72	Figure 1 – OCF Interaction.....	3
73	Figure 2 – User authorization and provisioning using Authorization Code Grant Flow	6
74	Figure 3 – Device Provisioning using Authorization Code Grant Flow.....	7
75	Figure 4 – Device connection with OCF Cloud	9
76		

77 **Tables**

78 Table 1 – Mapping of Properties of the "oic.r.account" and "oic.r.coapcloudconf"
79 Resources 8

80 Table 2 – Device connection with the OCF Cloud flow 10

81 Table 3 – Definition of the "oic.r.account" Resource..... 12

82 Table 4 – Properties of the "oic.r.account" Resource 12

83 Table 5 – Definition of the "oic.r.session" Resource 13

84 Table 6 – Properties of the "oic.r.session" Resource 13

85 Table 7 – Definition of the "oic.r.tokenrefresh" Resource 14

86 Table 8 – Properties of the "oic.r.tokenrefresh" Resource 14

87 Table 9 – Sensitive Data related to OCF Cloud 15

88 Table A.1 – Alphabetized list of security resources 16

89 Table A.2 – The Property definitions of the Resource with type "rt" = "oic.r.account". 19

90 Table A.3 – The CRUDN operations of the Resource with type "rt" = "oic.r.account". 20

91 Table A.4 – The Property definitions of the Resource with type "rt" = "oic.r.session". 22

92 Table A.5 – The CRUDN operations of the Resource with type "rt" = "oic.r.session". 23

93 Table A.6 – The Property definitions of the Resource with type "rt" = "oic.r.tokenrefresh". 26

94 Table A.7 – The CRUDN operations of the Resource with type "rt" = "oic.r.tokenrefresh". 27

95

96 **1 Purpose and Role**

97 This document defines security objectives, philosophy, resources and mechanism that impacts
98 OCF base layers of ISO/IEC 30118-1:2018. ISO/IEC 30118-1:2018 contains informative security
99 content. The OCF Security Document contains security normative content and may contain
100 informative content related to the OCF base or other OCF documents.

101 **2 Normative References**

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

105 IETF RFC 7228, *Terminology for Constrained-Node Networks*, May 2014,
106 <https://tools.ietf.org/html/rfc7228>

107 ISO/IEC 30118-1:2018 Information technology -- Open Connectivity Foundation (OCF) Document
108 -- Part 1: Core document
109 <https://www.iso.org/standard/53238.html>
110 Latest version available at:
111 https://openconnectivity.org/specs/OCF_Core_Specification.pdf

112 OCF Security Document, Information technology – Open Connectivity Foundation (OCF)
113 Document, Latest version available
114 at:https://openconnectivity.org/specs/OCF_Security_Specification.pdf

115 OCF Device to Cloud Services Document, Information technology – Open Connectivity
116 Foundation (OCF) Document – Part 8: Device to Cloud Services, Latest version available at:
117 https://openconnectivity.org/specs/OCF_OCF_Device_To_Cloud_Services_Specification.pdf

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

120 IETF RFC 6750, *The OAuth 2.0 Authorization Framework: Bearer Token Usage*, October 2012,
121 <https://tools.ietf.org/html/rfc6750>
122

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

125 oneM2M Release 3 Documents, <http://www.onem2m.org/technical/published-drafts>

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

128 **3 Terms, definitions, and abbreviated terms**

129 **3.1 Terms and definitions**

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

132 ISO and IEC maintain terminological databases for use in standardization at the following
133 addresses:

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

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

136 **3.1.1**
137 **Access Management Service (AMS)**
138 a service that dynamically constructs ACL Resources in response to a Device Resource request.

139 Note 1 to entry: An AMS can evaluate access policies remotely and supply the result to a Server which allows or denies
140 a pending access request. An AMS is authorised to provision ACL Resources.

141 **3.1.2**
142 **Trust Anchor**
143 a well-defined, shared authority, within a trust hierarchy, by which two cryptographic entities (e.g.
144 a Device and an onboarding tool) can assume trust

145 **3.1.3**
146 **OCF Security Domain**
147 a set of onboarded OCF Devices that are provisioned with credentialing information for confidential
148 communication with one another

149 **3.1.4**
150 **Access Token**
151 a credential used to authorize the connection with the OCF Cloud and access protected resources.
152 An Access Token is a string while the OCF Device has no internal logic based on its contents and
153 only forwards the token as-is

154 **3.1.5**
155 **Authorization Provider**
156 a Server issuing Access Tokens (3.1.4) to the Client after successfully authenticating the OCF
157 Cloud User (3.1.7) and obtaining authorization.

158 Note 1 to entry: Also known as authorization server in IETF RFC 6749.

159 **3.1.6**
160 **Device Registration**
161 a process by which Device is enrolled/registered to the OCF Cloud infrastructure (using Device
162 certificate and unique credential) and becomes ready for further remote operation through the cloud
163 interface (e.g. connection to remote Resources or publishing of its own Resources for access).

164 **3.1.7**
165 **OCF Cloud User**
166 a person or organization authorizing a set of Devices to interact with each other via an OCF Cloud.

167 Note 1 to entry: For each of the Devices, the OCF Cloud User is either the same as, or a delegate of, the person or
168 organization that onboarded that Device. The OCF Cloud User delegates, to the OCF Cloud authority, authority to route
169 between Devices registered by the OCF Cloud User. The OCF Cloud delegates, to the OCF Cloud User, authority to
170 select the set of Devices which can register and use the services of the OCF Cloud.

171 **3.2 Abbreviated terms**

172 **3.2.1**
173 **ACE**
174 Access Control Entry

175 **3.2.2**
176 **ACL**
177 Access Control List

178 **3.2.3**
179 **AMS**
180 Access Management Service

181 **3.2.4**
182 **CMS**
183 Credential Management Service

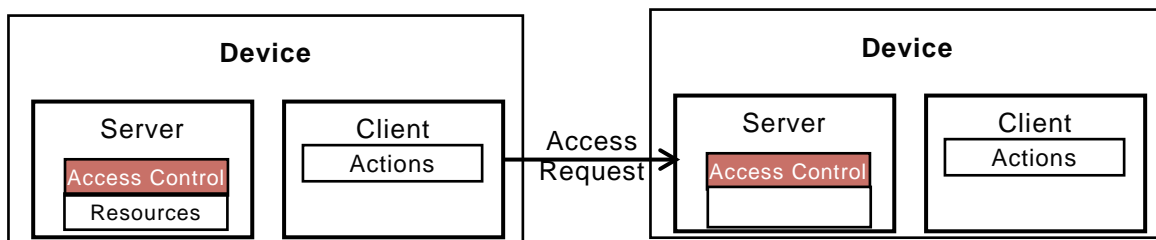
184 **4 Document Conventions and Organization**

185 **4.1 Conventions**

186 This document defines Resources, protocols and conventions used to implement security for OCF
187 core framework and applications.

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

189 Figure 1 depicts interaction between OCF Devices.



190

191

Figure 1 – OCF Interaction

192 Devices may implement a Client role that performs Actions on Servers. Actions access Resources
193 managed by Servers. The OCF stack enforces access policies on Resources. End-to-end Device
194 interaction can be protected using session protection protocol (e.g. DTLS) or with data encryption
195 methods.

196 **4.2 Notation**

197 In this document, features are described as required, recommended, allowed or DEPRECATED as
198 follows:

199 **Required** (or **shall** or **mandatory**).

200 These basic features shall be implemented to comply with OCF Core Architecture. The phrases
201 "shall not", and "PROHIBITED" indicate behavior that is prohibited, i.e. that if performed means the
202 implementation is not in compliance.

203 **Recommended** (or **should**).

204 These features add functionality supported by OCF Core Architecture and should be implemented.
205 Recommended features take advantage of the capabilities OCF Core Architecture, usually without
206 imposing major increase of complexity. Notice that for compliance testing, if a recommended
207 feature is implemented, it shall meet the specified requirements to be in compliance with these
208 guidelines. Some recommended features could become requirements in the future. The phrase
209 "should not" indicates behavior that is permitted but not recommended.

210 **Allowed** (may or allowed).

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

213 **Conditionally allowed** (CA)

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

216 **Conditionally required (CR)**

217 The definition or behaviour depends on a condition. If the specified condition is met, then the
218 definition or behaviour is required. Otherwise the definition or behaviour is allowed as default
219 unless specifically defined as not allowed.

220 **DEPRECATED**

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

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

228 Words that are emphasized are printed in italic.

229 **4.3 Data types**

230 See ISO/IEC 30118-1:2018.

231 **4.4 Document structure**

232 Informative clauses may be found in the Overview clauses, while normative clauses fall outside of
233 those clauses.

234 The Security Document may use the oneM2M Release 3 Documents,
235 <http://www.onem2m.org/technical/published-drafts>

236 OpenAPI as the API definition language. The mapping of the CRUDN actions is specified in
237 ISO/IEC 30118-1:2018.

238

239 **5 Security overview**

240 **5.1 Preamble**

241 A Device is authorized to communicate with an OCF Cloud if a trusted Mediator has provisioned
242 the Device.

- 243 – Device and Mediator connect over DTLS using "/oic/sec/cred"
- 244 – Device is provisioned by Mediator with following information:
 - 245 – the URL of OCF Cloud
 - 246 – Authorization Provider Name to identify the origin of the Access Token
 - 247 – Access Token / Authorization Code that is validated / exchanged by the OCF Cloud
 - 248 – UUID of the OCF Cloud

249 The OpenAPI 2.0 definitions (Annex A) used in this document are normative. This includes that all
250 defined payloads shall comply with the indicated OpenAPI 2.0 definitions. Annex A contains all of
251 the OpenAPI 2.0 definitions for Resource Types defined in this document.

252 **5.2 Device Provisioning for OCF Cloud and Device Registration Overview**

253 As mentioned in the start of Clause 0, communication between a Device and OCF Cloud is subject
254 to different criteria in comparison to Devices which are within a single local network. The Device is
255 configured in order to connect to the OCF Cloud by a Mediator as specified in the CoAPCloudConf
256 Resource clauses in OCF Device to Cloud Services . Provisioning includes the remote connectivity
257 and local details such as URL where the OCF Cloud hosting environment can be found, the OCF
258 Cloud verifiable Access Token and optionally the name of the Authorization Provider which issued
259 the Access Token.

260 NOTE a Device which connects to the OCF Cloud still retains the ownership established at onboarding with the DOTS.

261 **5.3 Credential overview**

262 Devices may use credentials to prove the identity and role(s) of the parties in bidirectional
263 communication

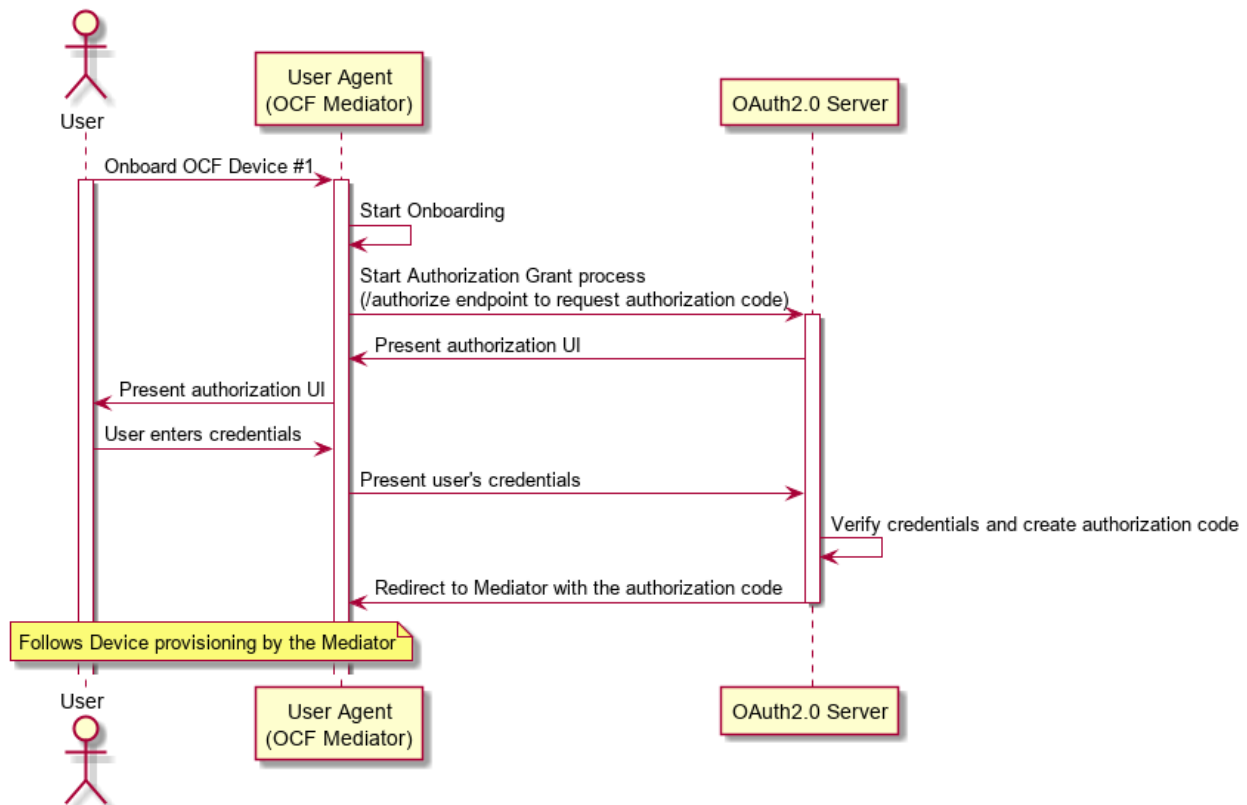
264 Access Tokens are provided to an OCF Cloud once an authenticated session with an OCF Cloud
265 is established, to verify the User ID with which the Device is to be associated.

266 **6 Device provisioning for OCF Cloud**

267 **6.1 Cloud Provisioning General**

268 The Device that connects to the OCF Cloud shall support the "oic.r.coapcloudconf" Resource on
269 Device and following SVRs on the OCF Cloud: "/oic/sec/account", "/oic/sec/session",
270 "/oic/sec/tokenrefresh".

271 The OCF Cloud is expected to use a secure mechanism for associating a Mediator with an OCF
272 Cloud User. The choice of mechanism is up to the OCF Cloud. Recommended solution is based on
273 the OAuth2.0 Authorization Grant Type flow specified in IETF RFC 6749, where the Mediator act
274 as a User-Agent and presents authorization UI to the user - see Figure 2. OCF Cloud is expected
275 to ensure that the suitable authentication mechanism is used to authenticate the OCF Cloud User.



276

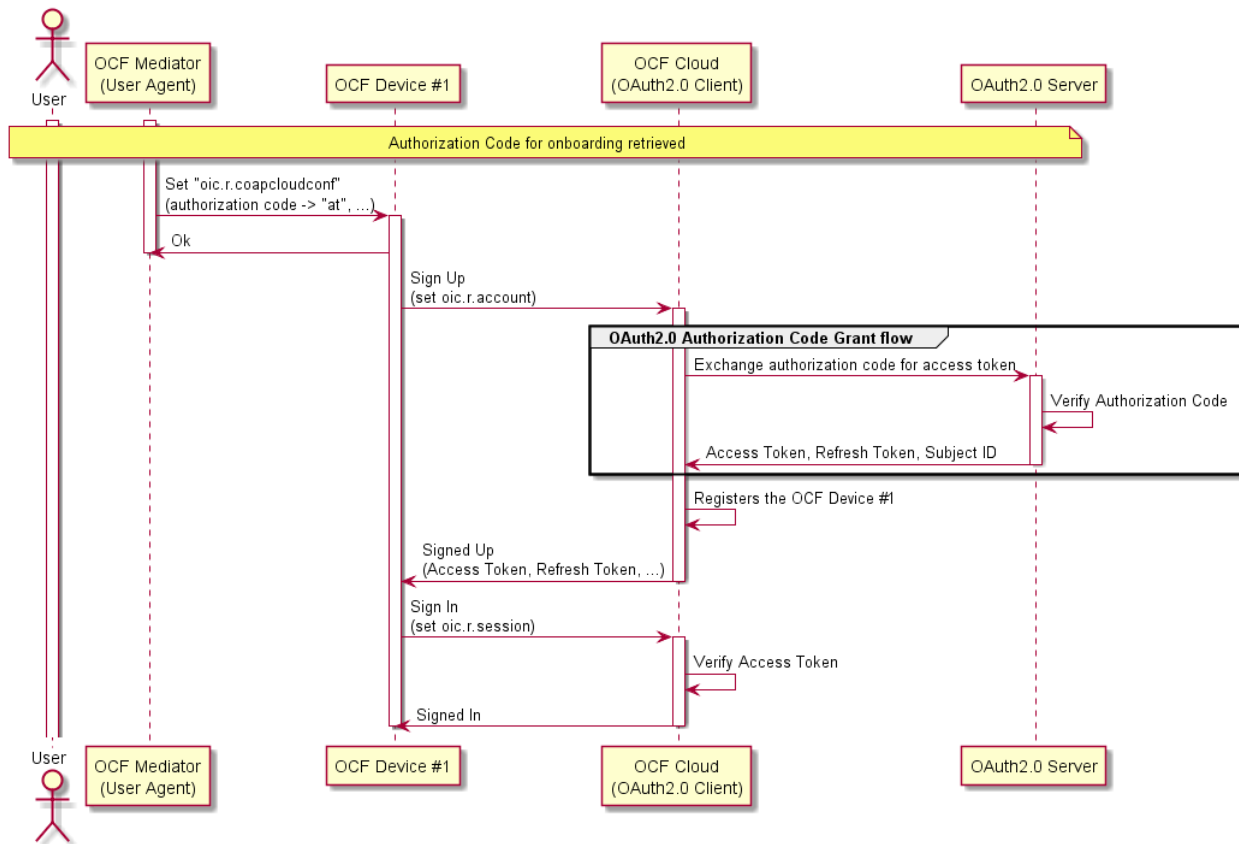
277 **Figure 2 – User authorization and provisioning using Authorization Code Grant Flow**

278 **6.2 Device Provisioning by Mediator**

279 The Mediator and the Device shall use the secure session to provision the Device to connect with
 280 the OCF Cloud.

281 The Mediator obtains an Authorization Code or directly an Access Token from the Authorization
 282 Server as described in OCF Device to Cloud Services . This value is then used by the Device for
 283 registering with the OCF Cloud as described in clause 7. At the time of Device Registration OCF
 284 Cloud exchanges the Authorization Code for the Access Token, returns it back to the OCF Device
 285 and associates the TLS session with corresponding Device ID. The OCF Cloud maintains a map
 286 where Access Token and Mediator provided Device ID are stored.

287 The Mediator provisions the Device, as described in OCF Device to Cloud Services . The Mediator
 288 provisions OCF Cloud URI to the "cis" Property of "oic.r.coapcloudconf" Resource, OCF Cloud
 289 UUID to the "sid" Property of "oic.r.coapcloudconf" Resource and per-device Access Token or
 290 Authorization Code to the "at" Property of "oic.r.coapcloudconf" Resource on Device. Exchanged
 291 and returned provisioned Access Token is to be treated by Device as an Access Token with
 292 "Bearer" token type as defined in IETF RFC 6750. The provisioned "at" value follows a proprietary
 293 data format, and may include multiple values marshalled/concatenated together into a single string
 294 (e.g. "{\"token\":\"abc\", \"client_id\":\"1234\", \"idp\":\"identityProvider1\"}") is a valid "at" Property
 295 value). See Figure 3 for the detailed overview of the recommended flow, which includes optional
 296 OAuth 2.0 Authorization Code Grant



297

298

Figure 3 – Device Provisioning using Authorization Code Grant Flow

299
300

For the purposes of access control, the Device shall identify the OCF Cloud using the OCF Cloud UUID in the Common Name field of the End-Entity certificate used to authenticate the OCF Cloud.

301
302

AMS should configure the ACE2 entries on a Device so that the Mediator(s) is the only Device(s) with UPDATE permission for the "oic.r.coapcloudconf" Resource.

303
304
305
306
307
308
309

The AMS should configure the ACE2 entries on the Device to allow request from the OCF Cloud. By request from the Mediator, the AMS removes old ACL2 entries with previous OCF Cloud UUID. This request happens before "oic.r.coapcloudconf" is configured by the Mediator for the new OCF Cloud. The Mediator also requests AMS to set the OCF Cloud UUID as the "subject" Property for the new ACL2 entries. AMS may use "sid" Property of "oic.r.coapcloudconf" Resource as the current OCF Cloud UUID. AMS could either provision a wildcard entry for the OCF Cloud or provision an entry listing each Resource published on the Device.

310
311
312

If OCF Cloud provides "redirecturi" Value as response during Device Registration, the redirected-to OCF Cloud is assumed to have the same OCF Cloud UUID and to use the same trust anchor. Otherwise, presented OCF Cloud UUID wouldn't match the provisioned ACL2 entries.

313
314
315
316
317
318
319

The Mediator should provision the "oic.r.coapcloudconf" Resource with the Properties in Table 1. These details once provisioned are used by the Device to perform Device Registration to the OCF Cloud. OCF Device is not expected to have any internal logic based on the values of "at" and "apn" Properties. The values of these Properties are forwarded as-is to the OCF Cloud. After the initial registration, the Device should use updated values received from the OCF Cloud instead. If OCF Cloud User wants the Device to re-register with the OCF Cloud, they can use the Mediator to re-provision the "oic.r.coapcloudconf" Resource with the new values.

320
321

Table 1 – Mapping of Properties of the "oic.r.account" and "oic.r.coapcloudconf" Resources

Property Title	oic.r.coapcloudconf	oic.r.account	Description
Authorization Provider Name	apn	authprovider	The name of Authorization Provider through which Access Token was obtained.
OCF Cloud URL	cis	-	This is the URL connection is established between Device and OCF Cloud.
Access Token	at	accesstoken	Access Token used to authorize the TLS connection for communication with the OCF Cloud, or the Authorization Code which is then verified and exchanged for the Access Token during Device Registration.
OCF Cloud UUID	sid	-	This is the identity of the OCF Cloud that the Device is configured to use.

322

7 Device authentication with OCF Cloud

323

7.1 Device Authentication with OCF Cloud General

324 The mechanisms for Device Authentication in clauses 10.2, 10.3 and 10.4 of OCF Security imply
325 that a Device is authorized to communicate with any other Device meeting the criteria provisioned
326 in "/oic/sec/cred"; the "/oic/sec/acl2" Resource (or "/oic/sec/acl1" resource of OIC1.1 Servers) are
327 additionally used to restrict access to specific Resources. The present clause describes Device
328 authentication for OCF Cloud, which uses slightly different criteria as described in clause 0. A
329 Device accessing an OCF Cloud shall establish a TLS session. The mutual authenticated TLS
330 session is established using Server certificate and Client certificate.

331 Each Device is identified by the Access Token obtained from the Device Registration response.
332 The OCF Cloud holds an OCF Cloud association table that maps Access Token, User ID and Device
333 ID. The Device Registration shall happen while the Device is in RFNOP state. After Device
334 Registration, the updated Access Token, Device ID and User ID are used by the Device for the
335 subsequent connection with the OCF Cloud.

7.2 Device Connection with the OCF Cloud

337 The Device should establish the TLS connection using the certificate based credential. The
338 connection should be established after Device is provisioned by Mediator.

339 The TLS session is established between Device and the OCF Cloud as specified in IETF RFC 8323.
340 The OCF Cloud is expected to provide certificate signed by trust anchor that is present in cred
341 entries of the Device. These cred entries are expected to be configured by the Mediator.

342 The Device shall validate the OCF Cloud's identity based on the credentials that are contained in
343 "/oic/sec/cred" Resource entries of the Device.

344 The OCF Cloud is expected to validate the manufacturer certificate provided by the Device.

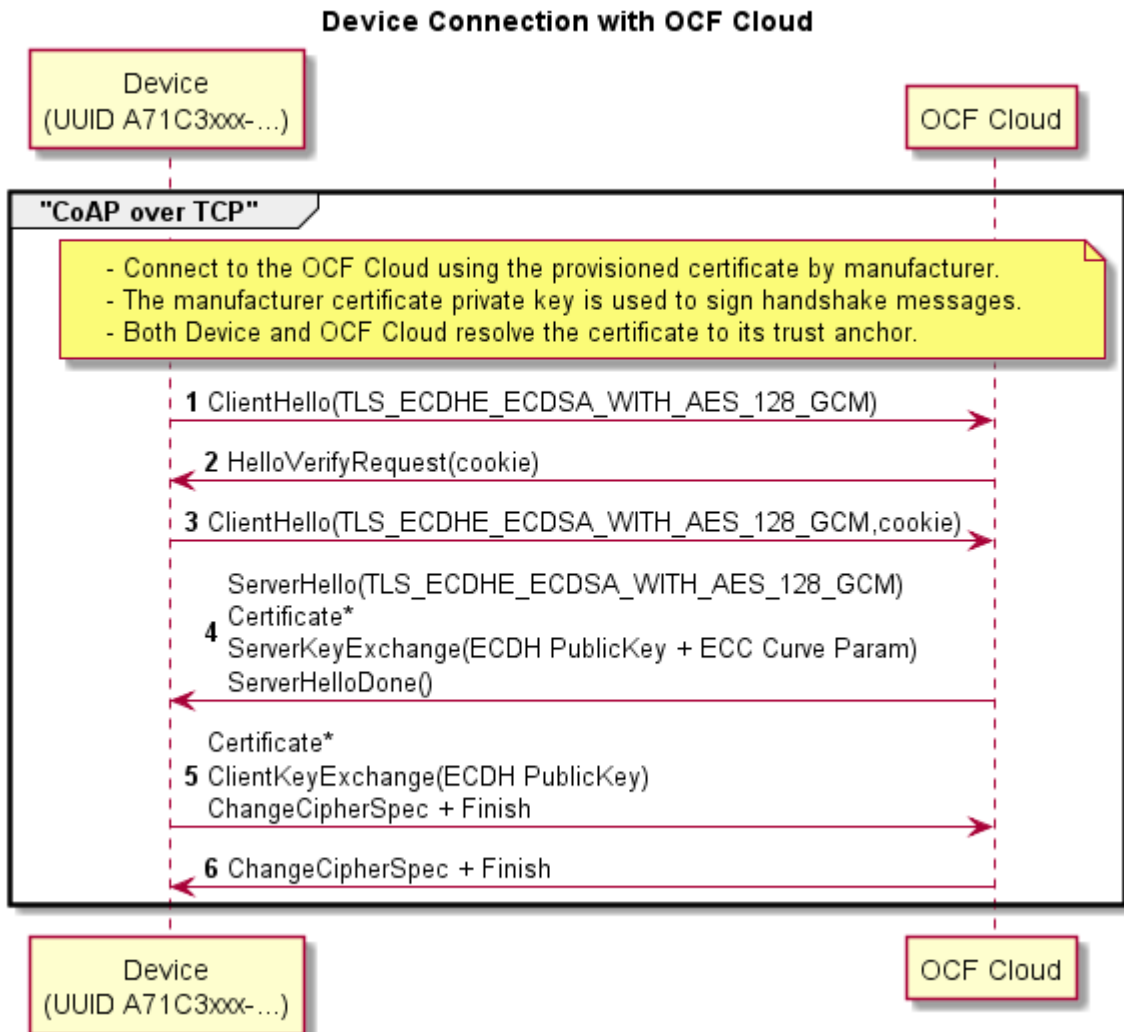
345 The assumption is that the OCF Cloud User trusts the OCF Cloud that the Device connects. The
346 OCF Cloud connection should not happen without the consent of the OCF Cloud User. The

347 assumption is that the OCF Cloud User has either service agreement with the OCF Cloud provider
348 or uses manufacturer provided OCF Cloud.

349 If authentication fails, the "clec" Property of "oic.r.coapcloudconf" Resource on the Device shall be
350 updated about the failed state, if it is supported by the Device. If authentication succeeds, the
351 Device and OCF Cloud should establish an encrypted link in accordance with the negotiated cipher
352 suite.

353 Figure 4 depicts sequence for Device connection with OCF Cloud and steps described in Table 2.

354 .



355

356

Figure 4 – Device connection with OCF Cloud

357

Table 2 – Device connection with the OCF Cloud flow

Steps	Description
1 - 6	TLS connection between the OCF Cloud and Device. The Device's manufacturer certificate may contain data attesting to the Device hardening and security properties

359 **7.3 Security Considerations**

360 When an OCF Server receives a request sent via the OCF Cloud, then the OCF Server permits
 361 that request using the identity of the OCF Cloud rather than the identity of the OCF Client. If there
 362 is no mechanism through which the OCF Cloud permits only those interactions which the user
 363 intends between OCF Clients and OCF Server via the OCF Cloud, and denies all other interactions,
 364 then OCF Clients might get elevated privileges by submitting a request via the OCF Cloud. This is
 365 highly undesirable from the security perspective. Consequently, OCF Cloud implementations are
 366 expected to provide some mechanism through which the OCF Cloud prevents OCF Clients getting
 367 elevated privileges when submitting a request via the OCF Cloud. In the present document release,
 368 the details of the mechanism are left to the implementation.

369 The security considerations about the manufacturer certificate as described in clause 7.3.6.5 of
 370 OCF Security are also applicable in the Device authentication with the OCF Cloud.

371 The Device should validate the OCF Cloud's TLS certificate as defined by IETF RFC 6125 and in
 372 accordance with its requirements for Server identity authentication.

373 The "uid" and "di" Property Value of "/oic/d" Resource may be considered personally identifiable
 374 information in some regulatory regions, and the OCF Cloud is expected to provide protections
 375 appropriate to its governing regulatory bodies.

376 **8 Message integrity and confidentiality**

377 **8.1 Cloud Session Semantics**

378 The messages between the OCF Cloud and Device shall be exchanged only if the Device and OCF
 379 Cloud authenticate each other as described in 7. The asymmetric cipher suites as described in 8.2
 380 shall be employed for establishing a secured session and for encrypting/decrypting between the
 381 OCF Cloud and the Device. The OCF Endpoint sending the message shall encrypt and authenticate
 382 the message using the cipher suite as described in 8.2 and the OCF Endpoint shall verify and
 383 decrypt the message before processing it.

384 **8.2 Cipher suites for OCF Cloud Credentials**

385 All Devices supporting OCF Cloud Certificate Credentials shall implement:

386 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256

387 All Devices supporting OCF Cloud Certificate Credentials should implement:

388 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,

389 TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256,

390 TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,

391 TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,

392 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

393

394 9 Security resources

395 9.1 Account Resource

396 The Account Resource specifies the Properties based on IETF RFC 6749 Access Token based
397 account creation. The mechanism to obtain credentials is described in Clause 6. The Account
398 Resource is used for Device Registration. The Account Resource is instantiated on the OCF Cloud
399 as "oic/sec/account" SVR and is used by cloud-enabled Devices to register with the OCF Cloud. It
400 should be only accessible on a secure channel; non-secure channel should not be able access this
401 Resource.

402 During the Device Registration process, an OCF Cloud can provide a distinct URI of another OCF
403 Cloud ("redirected-to" OCF Cloud). Both initial and redirected-to OCF Clouds are expected to
404 belong to the same Vendor; they are assumed to have the same UUID and are assumed to have
405 an Out-of-Band Communication Channel established. Device does not have to perform the Device
406 Registration on the redirected-to OCF Cloud and the OCF Cloud may ignore such attempts.
407 Redirected-to OCF Cloud is expected to accept the Access Token, provided to the Device by the
408 initial OCF Cloud.

409 The RETRIEVE operation on OCF Cloud's "/oic/sec/account" Resource is not allowed and the OCF
410 Cloud is expected to reject all attempts to perform such operation.

411 The UPDATE operation on the OCF Cloud's "/oic/sec/account" Resource behaves as follows:

- 412 – A Device intending to register with the OCF Cloud shall send UPDATE with following Properties
413 "di" ("di" Property Value of "/oic/d" Resource), and "accesstoken" as configured by the Mediator
414 ("at" Property Value of "oic.r.coapcloudconf" Resource). The OCF Cloud verifies it is the same
415 "accesstoken" which was assigned to the Mediator for the corresponding "di" Property Value.
416 The "accesstoken" is the permission for the Device to access the OCF Cloud. If the "apn" was
417 included when the Mediator UPDATED the "oic.r.coapcloudconf" Resource, the Device shall
418 also include "authprovider" Property when registering with the OCF Cloud. If no "apn" is
419 specified, then the "authprovider" Property shall not be included in the UPDATE request.
- 420 – OCF Cloud returns "accesstoken", "uid", "refreshtoken", and "expiresin" It may also return
421 "redirecturi". Received "accesstoken" is to be treated by Device as an Access Token with
422 "Bearer" token type as defined in IETF RFC 6750. This "accesstoken" shall be used for the
423 following Account Session start using "oic/sec/session" SVR. Received "refreshtoken" is to be
424 treated by Device as a Refresh Token as defined in IETF RFC 6749. The Device stores the
425 OCF Cloud's Response values. If "redirecturi" is received, Device shall use received value as
426 a new OCF Cloud URI instead of "cis" Property Value of "oic.r.coapcloudconf" Resource for
427 further connections.

428 The DELETE operation on the OCF Cloud's "/oic/sec/account" Resource should behave as follows:

- 429 – To deregister with the OCF Cloud, a DELETE operation shall be sent with the "accesstoken"
430 and either "uid", or "di" to be deregistered with the OCF Cloud. On DELETE with the OCF Cloud,
431 the Device should also delete values internally stored. Once deregister with an OCF Cloud,
432 Device can connect to any other OCF Cloud. Device deregistered need to go through the steps
433 in 6 again to be registered with the OCF Cloud.

434 Format of "oic.r.account" Resource is defined in Table 3.

435

Table 3 – Definition of the "oic.r.account" Resource

Fixed URI	Resource Type Title	Resource Type ID ("rt" value)	OCF Interfaces	Description	Related Functional Interaction
/oic/sec/account	Account	oic.r.account	oic.if.base line	Resource used for a device to add itself under a given credential	N/A

436 Table 4 defines the Properties of "oic.r.account ".

437

Table 4 – Properties of the "oic.r.account" Resource

Property Title	Property Name	Value Type	Value Rule	Access Mode	Mandatory	Description
Device ID	di	string	uuid	W	Yes	Unique Device identifier. Format pattern according to IETF RFC 4122.
Authorization Provider Name	authprovider	string	N/A	W	No	The name of Authorization Provider through which Access Token was obtained.
Access Token	accesstoken	string	Non-empty string	W	Yes	Access Token used to authorize and associate the TLS connection for communication with the OCF Cloud with the Device ID, or the Authorization Code which is then verified and exchanged for the Access Token during Device Registration.
Access Token	accesstoken	string	Non-empty string	R	Yes	Access Token used to authorize and associate the TLS connection for communication with the OCF Cloud with the Device ID.
Refresh Token	refresh token	string	Non-empty string	R	Yes	Refresh token can be used to refresh the Access Token before getting expired.
Token Expiration	expiresin	integer	-	R	Yes	Access Token life time in seconds (-1 if permanent).
User ID	uid	string	uuid	R	Yes	Unique OCF Cloud User identifier. Format pattern according to IETF RFC 4122.
Redirect URI	redirecturi	string	-	R	No	Using this URI, the Client needs to reconnect to a redirected OCF Cloud. If provided, this value shall be used by the Device instead of Mediator-provided URI during the Device Registration.

438 **9.2 Account Session resource**

439 The "/oic/sec/session" Resource hosted on the OCF Cloud is used for creating connections with
 440 the OCF Cloud subsequent to Device registration though "/oic/sec/account" Resource. The
 441 "/oic/sec/session" Resource requires the device ID, User ID and Access Token which are stored
 442 securely on the Device.

443 The "/oic/sec/session" Resource is exposed by the OCF Cloud. It should be only accessible on a
 444 secure channel; non-secure channel cannot access this Resource.

445 The RETRIEVE operation on OCF Cloud's "/oic/sec/session" Resource is not allowed and the OCF
 446 Cloud is expected to reject all attempts to perform such operation.

447 The UPDATE operation is defined as follows for OCF Cloud's "/oic/sec/session" Resource:

- 448 – The Device connecting to the OCF Cloud shall send an UPDATE request message to the OCF
 449 Cloud's "/oic/sec/session" Resource. The message shall include the "di" Property Value of

450 "/oic/d" Resource and "uid", "login" Value ("true" to establish connection; "false" to disconnect)
 451 and "accesstoken" as returned by OCF Cloud during Device Registration. The OCF Cloud
 452 verifies it is the same Access Token which was returned to the Device during Device
 453 Registration process or during Token Refresh. If Device was attempting to establish the
 454 connection and provided values were verified as correct by the OCF Cloud, OCF Cloud sends
 455 a response with remaining lifetime of the associated Access Token ("expiresin" Property Value).
 456 "oic.r.session" Resource is defined in Table 5.

457 **Table 5 – Definition of the "oic.r.session" Resource**

Fixed URI	Resource Type Title	Resource Type ID ("rt" value)	OCF Interfaces	Description	Related Functional Interaction
/oic/sec/session	Account Session	oic.r.session	oic.if.baseline	Resource that enables a device to manage its session using login or logout	N/A

458 Table 6 defines the Properties of "oic.r.session".

459 **Table 6 – Properties of the "oic.r.session" Resource**

Property Title	Property Name	Value Type	Value Rule	Access Mode	Mandatory	Description
User ID	uid	string	uuid	W	Yes	User ID provided by Device Registration process. Format pattern according to IETF RFC 4122.
Device ID	di	string	uuid	W	Yes	Unique device id registered for a Device.Format pattern according to IETF RFC 4122.
Access Token	accesstoken	string	A string of at least one character	W	Yes	Access Token used to authorize and associate the TLS connection for communication with the OCF Cloud with the Device ID
Login Status	login	boolean	N/A	W	Yes	Action for the request: true = login, false = logout
Token Expiration	expiresin	integer	N/A	R	Yes	Remaining Access Token life time in seconds (-1 if permanent) This Property is only provided to Device during connection establishment (when "login" Property Value equals "true"), it's not available otherwise

460 **9.3 Account Token Refresh Resource**

461 The "/oic/sec/tokenrefresh" Resource is used by the Device for refreshing the Access Token.

462 The "/oic/sec/tokenrefresh" Resource is hosted by the OCF Cloud. It should be only accessible on
 463 a secure channel; non-secure channel cannot access this Resource.

464 The Device should use "/oic/sec/tokenrefresh" to refresh the Access Token with the OCF Cloud,
 465 when the time specified in "expiresin" is near.

466 The RETRIEVE operation on OCF Cloud's "/oic/sec/ tokenrefresh" Resource is not allowed and the
 467 OCF Cloud is expected to reject all attempts to perform such operation.

468 The UPDATE operation is defined as follows for "/oic/sec/tokenrefresh" Resource

- 469 – The Device attempting to refresh the Access Token shall send an UPDATE request message
 470 to the OCF Cloud's "/oic/sec/tokenrefresh" Resource. The message shall include the "di"
 471 Property Value of "/oic/d" Resource, "uid" and "refreshtoken", as returned by OCF Cloud.
- 472 – OCF Cloud response is expected to include a "refreshtoken", new "acesstoken", and
 473 "expiresin". Received "acesstoken" is to be treated by Device as an Access Token with
 474 "Bearer" token type as defined in IETF RFC 6750. This Access Token is the permission for the
 475 Device to access the OCF Cloud. Received "refreshtoken" is to be treated by Device as a
 476 Refresh Token as defined in IETF RFC 6749. Received "refreshtoken" may be the new Refresh
 477 Token or the same one as provided by the Device in the UPDATE request. In case when new
 478 distinct "refreshtoken" is provided by the OCF Cloud, the Device shall discard the old value.
 479 The OCF Cloud's response values "refreshtoken", "acesstoken" and "expiresin" are securely
 480 stored on the Device.

481 "/oic.r.tokenrefresh" Resource is defined in Table 7.

482 **Table 7 – Definition of the "oic.r.tokenrefresh" Resource**

Fixed URI	Resource Type Title	Resource Type ID ("rt" value)	OCF Interfaces	Description	Related Functional Interaction
/oic/sec/tokenrefresh	Token Refresh	oic.r.tokenrefresh	oic.if.baseline	Resource to manage the access-token using refresh token	N/A

483 Table 8 defines the Properties of "oic.r.tokenrefresh".

484 **Table 8 – Properties of the "oic.r.tokenrefresh" Resource**

Property Title	Property Name	Value Type	Value Rule	Access Mode	Mandatory	Description
User ID	uid	string	uuid	W	Yes	User ID provided by Sign-up process. Format pattern according to IETF RFC 4122.
Device ID	di	string	uuid	W	Yes	Unique device id registered for an OCF Cloud User account. Format pattern according to IETF RFC 4122.
Refresh Token	refreshtoken	string	A string of at least one character	RW	Yes	Refresh token can be used to refresh the Access Token before getting expired.
Access Token	acesstoken	string	A string of at least one character	R	Yes	Access Token used to authorize and associate the TLS connection for communication with the OCF Cloud with the Device ID.
Token Expiration	expiresin	integer	-	R	Yes	Access Token life time in seconds (-1 if permanent).

485 10 Security hardening guidelines

486 10.1 Security hardening guidelines general

487 In addition to the Sensitive Data list outlined in Table 75 of Security Document, any Device
 488 implementing OCF Cloud connection capabilities should also provide reasonable protection for the
 489 information in Table 9.

490

Table 9 – Sensitive Data related to OCF Cloud

Data	Integrity protection	Confidentiality protection
OCF Cloud URL	Yes	Not required
OCF Cloud Identity	Yes	Not required

491

**Annex A
(normative)
Resource Type definitions**

492
493
494
495

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

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

Friendly Name (informative)	Resource Type (rt)	Clause
Account	oic.r.account	A.1
Account Session	oic.r.session	A.2
Account Token Refresh	oic.r.tokenrefresh	A.3

498 **A.1 Account Token**

499 **A.1.1 Introduction**

500 Sign-up using generic account provider.

501 **A.1.2 Well-known URI**

502 /oic/sec/account

503 **A.1.3 Resource type**

504 The Resource Type is defined as: "oic.r.account".

505 **A.1.4 OpenAPI 2.0 definition**

```

506 {
507   "swagger": "2.0",
508   "info": {
509     "title": "Account Token",
510     "version": "20190111",
511     "license": {
512       "name": "OCF Data Model License",
513       "url":
514 "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI
515 CENSE.md",
516       "x-copyright": "copyright 2016-2017, 2019 Open Connectivity Foundation, Inc. All rights
517 reserved."
518     },
519     "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
520   },
521   "schemes": ["http"],
522   "consumes": ["application/json"],
523   "produces": ["application/json"],
524   "paths": {
525     "/oic/sec/account" : {
526       "post": {
527         "description": "Sign-up using generic account provider.\n",
528         "parameters": [
529           { "$ref": "#/parameters/interface" },
530           {
531             "name": "body",
532             "in": "body",
533             "required": true,
534             "schema": { "$ref": "#/definitions/Account-request" },
535             "x-example":
536             {
537               "di" : "9cfbeb8e-5ale-4dlc-9d01-00c04fd430c8",
538               "authprovider" : "github",
539               "accesstoken" : "8802f2eaf8b5e147a936"
540             }
541           }
542         ]
543       }
544     }
545   }
546 }

```

```

541     }
542   ],
543   "responses": {
544     "204": {
545       "description": "2.04 Changed respond with required and optional information\n",
546       "x-example":
547         {
548           "rt": ["oic.r.account"],
549           "accesstoken" : "0f3d9f7fe5491d54077d",
550           "refreshtoken" : "00fe4644a6fbe5324eec",
551           "expiresin" : 3600,
552           "uid" : "123e4567-e89b-12d3-a456-d6e313b71d9f",
553           "redirecturi" : "coaps+tcp://example.com:443"
554         },
555       "schema": { "$ref": "#/definitions/Account-response" }
556     }
557   }
558 },
559 "delete": {
560   "description": "Delete a device. This also removes all resources in the device on cloud
561 side.\nexample: /oic/account?di=9cfbeb8e-5ale-4dlc-9d01-
562 00c04fd430c8&accesstoken=0f3d9f7fe5491d54077d\n",
563   "parameters": [
564     { "$ref": "#/parameters/interface" }
565   ],
566   "responses": {
567     "202": {
568       "description": "2.02 Deleted response informing the device is successfully
569 deleted.\n"
570     }
571   }
572 }
573 }
574 },
575 "parameters": {
576   "interface": {
577     "in": "query",
578     "name": "if",
579     "type": "string",
580     "enum": ["oic.if.baseline"]
581   }
582 },
583 "definitions": {
584   "Account-request": {
585     "properties": {
586       "authprovider": {
587         "description": "The name of Authorization Provider through which Access Token was
588 obtained",
589         "type": "string"
590       },
591       "accesstoken": {
592         "description": "Access-Token used for communication with OCF Cloud after account
593 creation",
594         "pattern": "(?!$|\\s+).*",
595         "type": "string"
596       },
597       "di": {
598         "description": "Format pattern according to IETF RFC 4122.",
599         "pattern": "^-[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
600 9]{12}$",
601         "type": "string"
602       }
603     },
604     "type": "object",
605     "required": ["di", "accesstoken"]
606   },
607   "Account-response": {
608     "properties": {
609       "expiresin": {
610         "description": "Access-Token remaining life time in seconds (-1 if permanent)",
611         "readOnly": true,

```

```

612         "type": "integer"
613     },
614     "rt": {
615         "description": "Resource Type of the Resource",
616         "items": {
617             "maxLength": 64,
618             "type": "string",
619             "enum" : ["oic.r.account"]
620         },
621         "minItems": 1,
622         "maxItems": 1,
623         "readOnly": true,
624         "type": "array"
625     },
626     "refreshToken" : {
627         "description": "Refresh token can be used to refresh the Access Token before getting
628 expired",
629         "pattern": "(?!$|\\s+).*",
630         "readOnly": true,
631         "type": "string"
632     },
633     "uid" : {
634         "description": "Format pattern according to IETF RFC 4122.",
635         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
636 9]{12}$",
637         "type": "string"
638     },
639     "accessToken" : {
640         "description": "Access-Token used for communication with cloud after account creation",
641         "pattern": "(?!$|\\s+).*",
642         "type": "string"
643     },
644     "n" : {
645         "$ref":
646 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
647 schema.json#/definitions/n"
648     },
649     "id" : {
650         "$ref":
651 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
652 schema.json#/definitions/id"
653     },
654     "redirecturi" : {
655         "description": "Using this URI, the Client needs to reconnect to a redirected OCF Cloud.
656 If provided, this value shall be used by the Device instead of Mediator-provided URI during the
657 Device Registration.",
658         "readOnly": true,
659         "type": "string"
660     },
661     "if" : {
662         "description": "The interface set supported by this resource",
663         "items": {
664             "enum": [
665                 "oic.if.baseline"
666             ],
667             "type": "string"
668         },
669         "minItems": 1,
670         "maxItems": 1,
671         "uniqueItems": true,
672         "readOnly": true,
673         "type": "array"
674     }
675 },
676 "type" : "object",
677 "required": ["accessToken", "refreshToken", "expiresin", "uid"]
678 }
679 }
680 }
681

```


682 **A.1.5 Property definition**

683 Table A.2 defines the Properties that are part of the "oic.r.account" Resource Type.

684 **Table A.2 – The Property definitions of the Resource with type "rt" = "oic.r.account".**

Property name	Value type	Mandatory	Access mode	Description
di	string	Yes	Write Only	Unique Device identifier. Format pattern according to IETF RFC 4122.
authprovider	string	No	Write Only	The name of Authorization Provider through which Access Token was obtained.
acesstoken	string	Yes	Write Only	Access Token used to authorize and associate the TLS connection for communication with the OCF Cloud with the Device ID, or the Authorization Code which is then verified and exchanged for the Access Token during Device Registration.
id	multiple types: see schema	No	Read Write	
refreshtoken	string	Yes	Read Only	Refresh token can be used to refresh the Access Token before getting expired.
rt	array: see schema	No	Read Only	Resource Type of the Resource
acesstoken	string	Yes	Read Only	Access Token used to authorize and associate the TLS connection for communication with the OCF Cloud with the Device ID.
uid	string	Yes	Read Only	Unique OCF Cloud User identifier. Format

				pattern according to IETF RFC 4122.
expiresin	integer	Yes	Read Only	Access-Token life time in seconds (-1 if permanent)
if	array: see schema	No	Read Only	The interface set supported by this resource
redirecturi	string	No	Read Only	Using this URI, the Client needs to reconnect to a redirected OCF Cloud. If provided, this value shall be used by the Device instead of Mediator-provided URI during the Device Registration.
n	multiple types: see schema	No	Read Write	

685 **A.1.6 CRUDN behaviour**

686 Table A.3 defines the CRUDN operations that are supported on the "oic.r.account" Resource Type.

687 **Table A.3 – The CRUDN operations of the Resource with type "rt" = "oic.r.account".**

Create	Read	Update	Delete	Notify
		post	delete	

688 **A.2 Session**

689 **A.2.1 Introduction**

690 Resource that manages the persistent session between a Device and OCF Cloud.

691 **A.2.2 Well-known URI**

692 /oic/sec/session

693 **A.2.3 Resource type**

694 The Resource Type is defined as: "oic.r.session".

695 **A.2.4 OpenAPI 2.0 definition**

```

696 {
697   "swagger": "2.0",
698   "info": {
699     "title": "Session",
700     "version": "v1.0-20181001",
701     "license": {
702       "name": "OCF Data Model License",
703       "url":
704 "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI
705 CENSE.md",
706       "x-copyright": "copyright 2016-2017, 2019 Open Connectivity Foundation, Inc. All rights
707 reserved."
708     },

```

```

709     "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
710   },
711   "schemes": ["http"],
712   "consumes": ["application/json"],
713   "produces": ["application/json"],
714   "paths": {
715     "/oic/sec/session" : {
716       "post": {
717         "description": "Resource that manages the persistent session between a Device and OCF
718 Cloud.",
719         "parameters": [
720           { "$ref": "#/parameters/interface" },
721           {
722             "name": "body",
723             "in": "body",
724             "required": true,
725             "schema": { "$ref": "#/definitions/Account-Session-Request" },
726             "x-example":
727               {
728                 "uid" : "123e4567-e89b-12d3-a456-d6e313b71d9f",
729                 "di" : "9cfbeb8e-5ale-4dlc-9d01-00c04fd430c8",
730                 "accesstoken" : "0f3d9f7fe5491d54077d",
731                 "login" : true
732               }
733           }
734         ],
735         "responses": {
736           "204": {
737             "description": "",
738             "x-example":
739               {
740                 "rt": ["oic.r.session"],
741                 "expiresin" : 3600
742               },
743             "schema": { "$ref": "#/definitions/Account-Session-Response" }
744           }
745         }
746       }
747     }
748   },
749   "parameters": {
750     "interface" : {
751       "in" : "query",
752       "name" : "if",
753       "type" : "string",
754       "enum" : ["oic.if.baseline"]
755     }
756   },
757   "definitions": {
758     "Account-Session-Request" : {
759       "properties": {
760         "uid": {
761           "description": "Format pattern according to IETF RFC 4122.",
762           "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
763 9]{12}$",
764           "type": "string"
765         },
766         "di": {
767           "description": "The Device ID\nFormat pattern according to IETF RFC 4122.",
768           "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
769 9]{12}$",
770           "type": "string"
771         },
772         "accesstoken": {
773           "description": "Access-Token used to grant access right for the Device to sign-in.",
774           "pattern": "(?!$|\\s+).*",
775           "type": "string"
776         },
777         "login": {
778           "description": "Action for the request: true = login, false = logout.",
779           "type": "boolean"

```

```

780     },
781     },
782     "type" : "object",
783     "required": ["uid", "di", "accesstoken", "login"]
784 },
785 "Account-Session-Response" : {
786     "properties": {
787         "expiresin": {
788             "description": "Access-Token remaining life time in seconds (-1 if permanent).",
789             "readOnly": true,
790             "type": "integer"
791         },
792         "rt": {
793             "description": "Resource Type of the Resource.",
794             "items": {
795                 "maxLength": 64,
796                 "type": "string",
797                 "enum": ["oic.r.session"]
798             },
799             "minItems": 1,
800             "readOnly": true,
801             "type": "array"
802         },
803         "n": {
804             "$ref":
805 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
806 schema.json#/definitions/n"
807         },
808         "id": {
809             "$ref":
810 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
811 schema.json#/definitions/id"
812         },
813         "if": {
814             "description": "The interface set supported by this Resource.",
815             "items": {
816                 "enum": [
817                     "oic.if.baseline"
818                 ],
819                 "type": "string"
820             },
821             "minItems": 1,
822             "readOnly": true,
823             "type": "array"
824         }
825     },
826     "type" : "object",
827     "required" : ["expiresin"]
828 }
829 }
830 }
831

```

832 A.2.5 Property definition

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

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

Property name	Value type	Mandatory	Access mode	Description
if	array: see schema	No	Read Only	The interface set supported by this Resource.
expiresin	integer	Yes	Read Only	Remaining Access Token life time in seconds (-1 if permanent). This Property is only

				provided to Device during connection establishment (when "login" Property Value equals "true"), it's not available otherwise.
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	
di	string	Yes	Write Only	Unique device id registered for a Device. Format pattern according to IETF RFC 4122.
accesstoken	string	Yes	Write Only	Access Token used to authorize and associate the TLS connection for communication with the OCF Cloud with the Device ID.
uid	string	Yes	Write Only	User ID provided by Device Registration process. Format pattern according to IETF RFC 4122.
login	boolean	Yes	Write Only	Action for the request: true = login, false = logout.

835 **A.2.6 CRUDN behaviour**

836 Table A.5 defines the CRUDN operations that are supported on the "oic.r.session" Resource Type.

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

Create	Read	Update	Delete	Notify
		post		

838 **A.3 Token Refresh**

839 **A.3.1 Introduction**

840 Obtain fresh Access Token using the refresh token, client should refresh Access Token before it
841 expires.

842 **A.3.2 Well-known URI**

843 /oic/sec/tokenrefresh

844 **A.3.3 Resource type**

845 The Resource Type is defined as: "oic.r.tokenrefresh".

846 **A.3.4 OpenAPI 2.0 definition**

```
847 {
848   "swagger": "2.0",
849   "info": {
850     "title": "Token Refresh",
851     "version": "v1.0-20181001",
852     "license": {
853       "name": "OCF Data Model License",
854       "url":
855         "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI
856         CENSE.md",
857       "x-copyright": "copyright 2016-2017, 2019 Open Connectivity Foundation, Inc. All rights
858         reserved."
859     },
860     "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
861   },
862   "schemes": ["http"],
863   "consumes": ["application/json"],
864   "produces": ["application/json"],
865   "paths": {
866     "/oic/sec/tokenrefresh" : {
867       "post": {
868         "description": "Obtain fresh access-token using the refresh token, client should refresh
869         access-token before it expires.\n",
870         "parameters": [
871           { "$ref": "#/parameters/interface" },
872           {
873             "name": "body",
874             "in": "body",
875             "required": true,
876             "schema": { "$ref": "#/definitions/TokenRefresh-Request" },
877             "x-example":
878               {
879                 "uid" : "123e4567-e89b-12d3-a456-d6e313b71d9f",
880                 "di" : "9cfbeb8e-5ale-4dlc-9d01-00c04fd430c8",
881                 "refreshToken" : "00fe4644a6fbe5324eec"
882               }
883           }
884         ],
885         "responses": {
886           "204": {
887             "description": "2.04 Changed respond with new access-token.\n",
888             "x-example":
889               {
890                 "rt": ["oic.r.tokenrefresh"],
891                 "accesstoken" : "8ce598980761869837be",
892                 "refreshToken" : "d4922312b6df0518e146",
893                 "expiresin" : 3600
894               }
895             ,
896             "schema": { "$ref": "#/definitions/TokenRefresh-Response" }
897           }
898         }
899       }
900     }
901   },
902   "parameters": {
903     "interface" : {
904       "in" : "query",
905       "name" : "if",
906       "type" : "string",
907       "enum" : ["oic.if.baseline"]
908     }
909   }
910 }
```

```

909     },
910     "definitions": {
911         "TokenRefresh-Request" : {
912             "properties": {
913                 "refreshToken": {
914                     "description": "Refresh token received by account management or during token refresh
915 procedure.",
916                     "pattern": "(?!$|\\s+).*",
917                     "type": "string"
918                 },
919                 "uid": {
920                     "description": "Format pattern according to IETF RFC 4122.",
921                     "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
922 9]{12}$",
923                     "type": "string"
924                 },
925                 "di": {
926                     "description": "Format pattern according to IETF RFC 4122.",
927                     "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
928 9]{12}$",
929                     "type": "string"
930                 }
931             },
932             "type": "object",
933             "required": ["uid", "di", "refreshToken"]
934         },
935         "TokenRefresh-Response" : {
936             "properties": {
937                 "expiresin": {
938                     "description": "Access-Token life time in seconds (-1 if permanent).",
939                     "readOnly": true,
940                     "type": "integer"
941                 },
942                 "rt": {
943                     "description": "Resource Type of the Resource.",
944                     "items": {
945                         "maxLength": 64,
946                         "type": "string",
947                         "enum": ["oic.r.tokenrefresh"]
948                     },
949                     "minItems": 1,
950                     "readOnly": true,
951                     "type": "array"
952                 },
953                 "refreshToken": {
954                     "description": "Refresh token received by account management or during token refresh
955 procedure.",
956                     "pattern": "(?!$|\\s+).*",
957                     "type": "string"
958                 },
959                 "accesstoken": {
960                     "description": "Granted Access-Token.",
961                     "pattern": "(?!$|\\s+).*",
962                     "readOnly": true,
963                     "type": "string"
964                 },
965                 "n": {
966                     "$ref":
967 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
968 schema.json#/definitions/n"
969                 },
970                 "id": {
971                     "$ref":
972 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
973 schema.json#/definitions/id"
974                 },
975                 "if" :
976                 {
977                     "description": "The interface set supported by this Resource.",
978                     "items": {
979                         "enum": [

```

```

980         "oic.if.baseline"
981     ],
982     "type": "string"
983 },
984 "minItems": 1,
985 "readOnly": true,
986 "type": "array"
987 }
988 },
989 "type" : "object",
990 "required": ["accesstoken", "refreshtoken", "expiresin"]
991 }
992 }
993 }
994

```

995 A.3.5 Property definition

996 Table A.6 defines the Properties that are part of the "oic.r.tokenrefresh" Resource Type.

997 **Table A.6 – The Property definitions of the Resource with type "rt" = "oic.r.tokenrefresh".**

Property name	Value type	Mandatory	Access mode	Description
refreshtoken	string	Yes	Write Only	Refresh token can be used to refresh the Access Token before getting expired.
uid	string	Yes	Write Only	User ID provided by Sign-up process. Format pattern according to IETF RFC 4122.
di	string	Yes	Write Only	Unique device id registered for an OCF Cloud User account. Format pattern according to IETF RFC 4122.
if	array: see schema	No	Read Only	The interface set supported by this Resource.
expiresin	integer	Yes	Read Only	Access Token life time in seconds (-1 if permanent).
accesstoken	string	Yes	Read Only	Access Token used to authorize and associate the TLS connection for communication with the OCF Cloud with the Device ID.
refreshtoken	string	Yes	Read Only	Refresh token can be used to

				refresh the Access Token before getting expired.
n	multiple types: see schema	No	Read Write	
rt	array: see schema	No	Read Only	Resource Type of the Resource.
id	multiple types: see schema	No	Read Write	

998 **A.3.6 CRUDN behaviour**

999 Table A.7 defines the CRUDN operations that are supported on the "oic.r.tokenrefresh" Resource
 1000 Type.

1001 **Table A.7 – The CRUDN operations of the Resource with type "rt" = "oic.r.tokenrefresh".**

Create	Read	Update	Delete	Notify
		post		

1002

1003