

OCF Cloud Security Specification

VERSION 2.2.4 | August 2021



OPEN CONNECTIVITY
FOUNDATION™

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

LEGAL DISCLAIMER

1
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 © 2019-2021 Open Connectivity Foundation, Inc. All rights reserved.

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

CONTENTS

17		
18		
19	Introduction.....	vi
20	1 Scope.....	1
21	2 Normative References	1
22	3 Terms, definitions and abbreviated terms	2
23	3.1 Terms and definitions.....	2
24	3.2 Abbreviated terms.....	2
25	4 Document Conventions and Organization	3
26	4.1 Conventions.....	3
27	4.2 Notation.....	3
28	4.3 Data types	4
29	5 Security overview	5
30	5.1 Preamble	5
31	5.2 OCF Cloud architecture alignment with ISO IEC 17789	5
32	5.3 Device provisioning for OCF Cloud and Device registration overview	6
33	5.4 Credential overview	6
34	6 Device provisioning for OCF Cloud	6
35	6.1 OCF Cloud provisioning general	6
36	6.2 Device provisioning by Mediator	7
37	6.3 Device Deregistration from the OCF Cloud by Mediator	9
38	7 Device authentication with OCF Cloud.....	10
39	7.1 Device authentication with OCF Cloud general	10
40	7.2 Device connection with the OCF Cloud	10
41	7.3 Security considerations.....	12
42	8 Message integrity and confidentiality	13
43	8.1 OCF Cloud session semantics	13
44	8.2 Cipher suites for OCF Cloud Credentials	13
45	9 Security Resources	13
46	9.1 Account Resource.....	13
47	9.2 Account Session Resource	15
48	9.3 Account Token Refresh Resource	16
49	10 Security hardening guidelines.....	17
50	10.1 Security hardening guidelines general	17
51	Annex A (normative) Resource Type definitions	18
52	A.1 List of Resource Type definitions	18
53	A.2 Account Token.....	18
54	A.2.1 Introduction	18
55	A.2.2 Well-known URI.....	18
56	A.2.3 Resource type	18
57	A.2.4 OpenAPI 2.0 definition.....	18
58	A.2.5 Property definition	21
59	A.2.6 CRUDN behaviour	22
60	A.3 Session.....	22

61	A.3.1	Introduction	22
62	A.3.2	Well-known URI	22
63	A.3.3	Resource type	22
64	A.3.4	OpenAPI 2.0 definition.....	22
65	A.3.5	Property definition	24
66	A.3.6	CRUDN behaviour	25
67	A.4	Token Refresh	25
68	A.4.1	Introduction	25
69	A.4.2	Well-known URI	25
70	A.4.3	Resource type	26
71	A.4.4	OpenAPI 2.0 definition.....	26
72	A.4.5	Property definition	28
73	A.4.6	CRUDN behaviour	29
74			

FIGURES

75	
76	Figure 1 – User authorization and provisioning using Authorization Code Grant Flow7
77	Figure 2 – Device provisioning using Authorization Code Grant Flow8
78	Figure 3 – Device deregistration from a Mediator flow.....10
79	Figure 4 – Device connection with OCF Cloud12
80	

81		Tables	
82	Table 1 – Mapping of Properties of "oic.r.account" and "oic.r.coapcloudconf" Resources		9
83	Table 2 – Device connection with the OCF Cloud flow		12
84	Table 3 – Definition of the "oic.r.account" Resource		14
85	Table 4 – Properties of the "oic.r.account" Resource		15
86	Table 5 – Definition of the "oic.r.session" Resource		16
87	Table 6 – Properties of the "oic.r.session" Resource		16
88	Table 7 – Definition of the "oic.r.tokenrefresh" Resource		17
89	Table 8 – Properties of the "oic.r.tokenrefresh" Resource		17
90	Table 9 – Sensitive Data related to OCF Cloud		17
91	Table A.1 – Alphabetized list of security Resources		18
92	Table A.2 – The Property definitions of the Resource with type "rt" = "oic.r.account".		21
93	Table A.3 – The CRUDN operations of the Resource with type "rt" = "oic.r.account".		22
94	Table A.4 – The Property definitions of the Resource with type "rt" = "oic.r.session".		24
95	Table A.5 – The CRUDN operations of the Resource with type "rt" = "oic.r.session".		25
96	Table A.6 – The Property definitions of the Resource with type "rt" = "oic.r.tokenrefresh".		28
97	Table A.7 – The CRUDN operations of the Resource with type "rt" = "oic.r.tokenrefresh".....		29
98			
99			
100			

101 **Introduction**

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

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

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

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

120 The OCF specification suite is made up of nineteen discrete documents, the documents fall into
121 logical groupings as described herein:

- 122 – Core framework
 - 123 – Core Specification
 - 124 – Security Specification
 - 125 – Onboarding Tool Specification
- 126 – Bridging framework and bridges
 - 127 – Bridging Specification
 - 128 – Resource to Alljoyn Interface Mapping Specification
 - 129 – OCF Resource to oneM2M Resource Mapping Specification
 - 130 – OCF Resource to BLE Mapping Specification
 - 131 – OCF Resource to EnOcean Mapping Specification
 - 132 – OCF Resource to LWM2M Mapping Specification
 - 133 – OCF Resource to UPlus Mapping Specification
 - 134 – OCF Resource to Zigbee Cluster Mapping Specification
 - 135 – OCF Resource to Z-Wave Mapping Specification
- 136 – Resource and Device models
 - 137 – Resource Type Specification
 - 138 – Device Specification
- 139 – Core framework extensions
 - 140 – Easy Setup Specification
 - 141 – Core Optional Specification
- 142 – OCF Cloud
 - 143 – Cloud API for Cloud Services Specification

- 144 – Device to Cloud Services Specification
- 145 – Cloud Security Specification

Cloud Security Specification

146

147 **1 Scope**

148 The OCF Cloud specifications are divided into a series of documents:

- 149 – OCF Cloud security specification (this document): The cloud security specification document
150 specifies the security requirements and definitions for OCF devices and OCF clouds
151 implementations.
- 152 – OCF Device to Cloud Specification: The OCF Device to Cloud Specification document defines
153 functional extensions and capabilities to meet the requirements of the OCF Cloud. This document
154 specifies new Resource Types to enable the functionality and any extensions required to connect an
155 OCF device to an OCF cloud.
- 156 – OCF Cloud API for cloud services specification: The Cloud API for cloud services specification
157 defines the OCF cloud API.

158 **2 Normative References**

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

162 ISO/IEC 30118-1 *Information technology – Open Connectivity Foundation (OCF) Document – Part*
163 *1: Core specification*

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

165 Latest version available at:

166 https://openconnectivity.org/specs/OCF_Core_Specification.pdf

167 ISO/IEC 30118-2, *Information technology – Open Connectivity Foundation (OCF) Document –*
168 *Part 2: Security specification*

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

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

171 ISO/IEC 30118-8, *Information technology – Open Connectivity Foundation (OCF) Document –*
172 *Part 8: Device to Cloud Services,*

173 <https://www.iso.org/standard/79360.html>

174 Latest version available at:

175 https://openconnectivity.org/specs/OCF_OCF_Device_To_Cloud_Services_Specification.pdf

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

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

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

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

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

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

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

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

184

185 IETF RFC 8323, *CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets*,

186 February 2018, <https://tools.ietf.org/html/rfc8323>

187 OCF Device to Cloud Services, *Open Connectivity Foundation Device to Cloud Services*
188 *Specification, Version 2.2.0*

189 Available at:
190 https://openconnectivity.org/specs/OCF_Device_To_Cloud_Services_Specification_v2.2.0.pdf
191 Latest version available at:
192 https://openconnectivity.org/specs/OCF_Device_To_Cloud_Services_Specification.pdf

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

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

196 **3 Terms, definitions and abbreviated terms**

197 **3.1 Terms and definitions**

198 For the purposes of this document, the terms and definitions given in ISO/IEC 30118-1, ISO/IEC
199 30118-2, ISO/IEC 30118-8 and the following apply.

200 ISO and IEC maintain terminological databases for use in standardization at the following
201 addresses:

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

204 **3.1.1**

205 **Access Token**

206 credential used to authorize the connection with the OCF Cloud and access protected Resources

207 Note 1 to entry: An Access Token is a string while the OCF Device has no internal logic based on its contents and
208 only forwards the token as-is

209 **3.1.2**

210 **Authorization Provider**

211 server issuing Access Tokens (3.1.1) via a Mediator to the Client after successfully authenticating
212 the *OCF Cloud User* (3.1.4) and obtaining authorization

213 Note 1 to entry: Also known as authorization server in ISO/IEC 17788 *Information technology –*
214 *Cloud computing – Overview and vocabulary*
215 <https://www.iso.org/standard/60544.html>

216 ISO/IEC 17789 *Information technology – Cloud computing – Reference architecture*
217 <https://www.iso.org/standard/60545.html>

218 IETF RFC 6749.

219 **3.1.3**

220 **Device Registration**

221 process by which Device is enrolled/registered to the OCF Cloud infrastructure (using Device
222 certificate and unique credential) and becomes ready for further remote operation through the cloud
223 interface (e.g. connection to remote Resources or publishing of its own Resources for access)

224 **3.1.4**

225 **OCF Cloud User**

226 person or organization authorizing a set of Devices to interact with each other via an OCF Cloud

227 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
228 organization that onboarded that Device. The OCF Cloud User delegates, to the OCF Cloud authority, authority to route
229 between Devices registered by the OCF Cloud User. The OCF Cloud delegates, to the OCF Cloud User, authority to
230 select the set of Devices which can register and use the services of the OCF Cloud.

231 **3.2 Abbreviated terms**

232 For the purposes of this document, the symbols and abbreviated terms given in ISO/IEC 30118-1,
233 ISO/IEC 30118-2 and ISO/IEC 30118-8 apply.

234 4 Document Conventions and Organization

235 4.1 Conventions

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

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

243 4.2 Notation

244 In this document, features are described as required, recommended, allowed or DEPRECATED as
245 follows:

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

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

250 Recommended (or should)(S).

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

257 Allowed (may or allowed)(O).

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

261 DEPRECATED.

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

268 Conditionally allowed (CA).

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

271 Conditionally required (CR).

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

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

276 Words that are emphasized are printed in italic.

277 In all of the Property and Resource definition tables that are included throughout this document the
278 "Mandatory" column indicates that the item detailed is mandatory to implement; the mandating of
279 inclusion of the item in a Resource Payload associated with a CRUDN action is dependent on the
280 applicable schema for that action.

281 **4.3 Data types**

282 Resources are defined using data types derived from JSON values as defined in clause 4.3 in
283 ISO/IEC 30118-1

284 **5 Security overview**

285 **5.1 Preamble**

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

- 288 – Device and Mediator connect over DTLS using "/oic/sec/cred"
- 289 – Device is provisioned by Mediator with following information:
 - 290 – the URL of OCF Cloud
 - 291 – Authorization Provider Name to identify the origin of the Access Token
 - 292 – Access Token / Authorization Code that is validated / exchanged by the OCF Cloud
 - 293 – UUID of the OCF Cloud

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

297 **5.2 OCF Cloud architecture alignment with ISO IEC 17789**

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

302 OCF defines an application capabilities type cloud service, providing Communication as a Service
303 (CaaS) (reference ISO/IEC 17788). This cloud service is provided by a cloud service provider, the
304 mechanisms used by the cloud service provider in managing their overall cloud infrastructure are
305 outside the scope of the OCF defined cloud service. The OCF definition is specific to the interface
306 offered by the cloud service to the cloud service customer, specifically the cloud service user.
307

308 There are three different user views defined. In the case where the cloud service customer is an
309 OCF Device as specified in OCF Device to Cloud Services then the views provided are:

- 310 - Interface for the OCF Device to provide information to the cloud service
- 311 - Interface for the OCF Device to retrieve information that has been provided to the cloud
312 service

313
314 In the case where the cloud service customer is another instance of a cloud service as specified in
315 this document then the view provided is:

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

318
319 The OCF Cloud service pertains specifically to a cloud service user, there is a single applicable
320 cloud service activity, that of "Use cloud service" defined in clause 8.2.21 of ISO/IEC 17789 .
321

322 Credentials for the user of the cloud service are provided using OAUTH2.0 as defined by RFC 6749.
323 The cloud service, either itself, or leveraging an external authorization server, provides a bearer
324 token that is required in all requests from all cloud users. Please see clause 7 of this document.
325

326 All connectivity between a cloud user and the OCFCloud service is via mutually authenticated TLS;
327 see clause 7.1 of this document.
328

329 ISO/IEC 27017 defines a code of practice for organizational level information security controls, and
330 implementation guidance for cloud services. Implementation and organizational level controls are
331 out of scope of the OCF Cloud Security Specification.

332 **5.3 Device provisioning for OCF Cloud and Device registration overview**

333 As mentioned in the start of Clause 0, communication between a Device and OCF Cloud is subject
334 to different criteria in comparison to Devices which are within a single local network. The Device is
335 configured in order to connect to the OCF Cloud by a Mediator as specified in the CoAPCloudConf
336 Resource clauses in ISO/IEC 30118-8. Provisioning includes the remote connectivity and local
337 details such as URL where the OCF Cloud hosting environment can be found, the OCF Cloud
338 verifiable Access Token and optionally the name of the Authorization Provider which issued the
339 Access Token.

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

341 **5.4 Credential overview**

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

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

346 **6 Device provisioning for OCF Cloud**

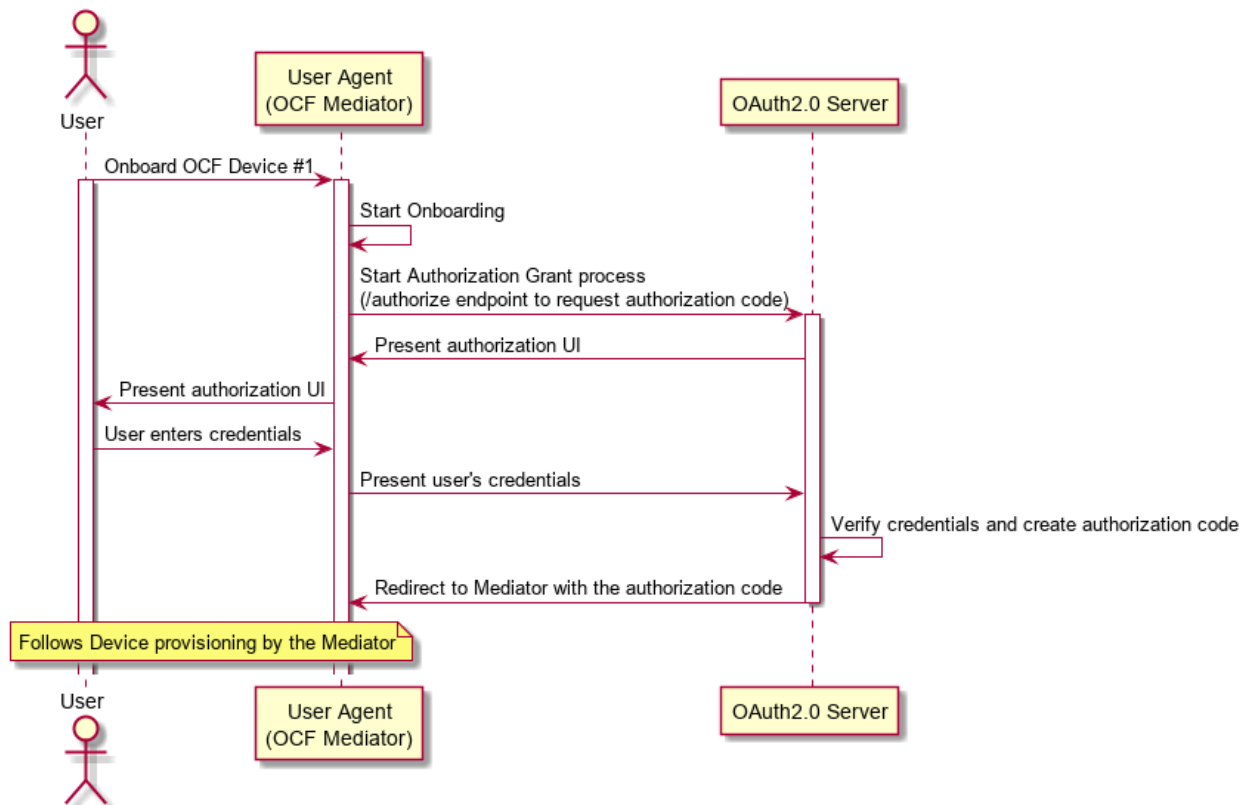
347 **6.1 OCF Cloud provisioning general**

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

351 The OCF Cloud is expected to use a secure mechanism for associating a Mediator with an OCF
352 Cloud User. The choice of mechanism is up to the OCF Cloud. Recommended solution is based on
353 the OAuth2.0 Authorization Grant Type flow specified in ISO/IEC 17788 *Information technology –*
354 *Cloud computing – Overview and vocabulary*
355 <https://www.iso.org/standard/60544.html>

356 ISO/IEC 17789 *Information technology – Cloud computing – Reference architecture*
357 <https://www.iso.org/standard/60545.html>

358 IETF RFC 6749, where the Mediator act as a User-Agent and presents authorization UI to the user
359 - see Figure 1. OCF Cloud is expected to ensure that the suitable authentication mechanism is
360 used to authenticate the OCF Cloud User.



361

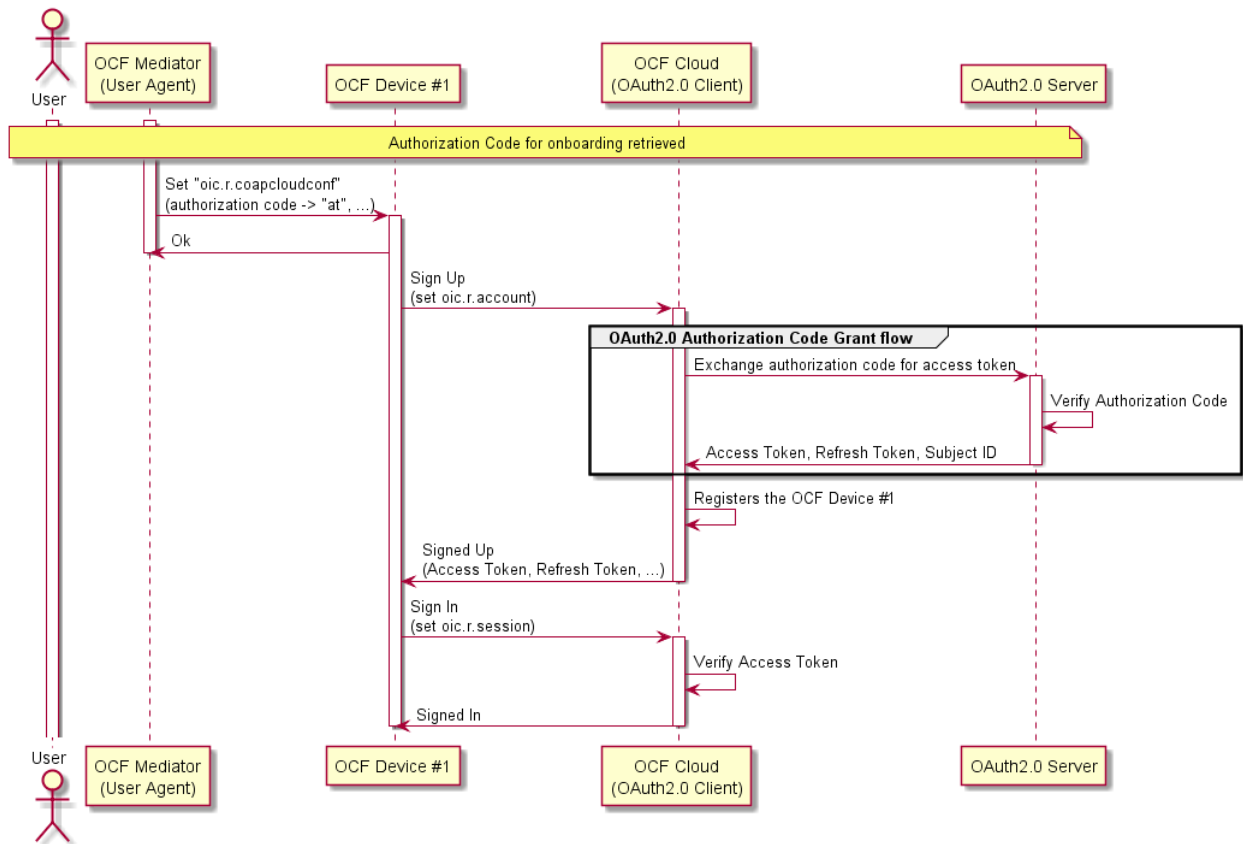
362 **Figure 1 – User authorization and provisioning using Authorization Code Grant Flow**

363 **6.2 Device provisioning by Mediator**

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

366 The Mediator obtains an Authorization Code or directly an Access Token from the Authorization
 367 Server as described in ISO/IEC 30118-8. This value is then used by the Device for registering with
 368 the OCF Cloud as described in clause 6.3. At the time of Device Registration OCF Cloud exchanges
 369 the Authorization Code for the Access Token, returns it back to the OCF Device and associates
 370 the TLS session with corresponding Device UUID. The OCF Cloud maintains a map where Access
 371 Token and Mediator provided Device UUID are stored.

372 The Mediator provisions the Device, as described in ISO/IEC 30118-8. The Mediator provisions
 373 OCF Cloud URI to the "cis" Property of "oic.r.coapcloudconf" Resource, OCF Cloud UUID to the
 374 "sid" Property of "oic.r.coapcloudconf" Resource and per-Device Access Token or Authorization
 375 Code to the "at" Property of "oic.r.coapcloudconf" Resource on Device. Exchanged and returned
 376 provisioned Access Token is to be treated by Device as an Access Token with "Bearer" token type
 377 as defined in IETF RFC 6750. The provisioned "at" value follows a proprietary data format, and
 378 may include multiple values marshalled/concatenated together into a single string (e.g.
 379 "{\"token\": \"abc\", \"client_id\": \"1234\", \"idp\": \"identityProvider1\"}" is a valid "at" Property value).
 380 See Figure 2 for the detailed overview of the recommended flow, which includes optional OAuth
 381 2.0 Authorization Code Grant



382

383

Figure 2 – Device provisioning using Authorization Code Grant Flow

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

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

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

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

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

405 **Table 1 – Mapping of Properties of "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.

406 **6.3 Device Deregistration from the OCF Cloud by Mediator**

407 To deregister a Device from the OCF Cloud without resetting the Device to manufacturer defaults,
 408 a Mediator with UPDATE permission for the "oic.r.coapcloudconf" Resource shall use a secure
 409 session for the following steps.

410 The Mediator shall send an UPDATE request with the empty "cis" Property to the
 411 "oic.r.coapcloudconf" Resource. The Device shall return a successful response followed by the
 412 DELETE request sent to the OCF Cloud's "/oic/sec/account" Resource it is connected to in order
 413 to deregister from the OCF Cloud as defined in clause 6.1. The Device shall also reset the
 414 "oic.r.coapcloudconf" Properties back to the default values. Please see Figure 3 for an illustration
 415 of this sequence.

416 Upon receiving the success response to the UPDATE request from the Device, the Mediator shall
 417 remove all of the ACE2 entries with the OCF Cloud UUID from the Device and shall also remove
 418 any credentials used to validate the OCF Cloud's identity that are contained in the "/oic/sec/cred"
 419 Resource of the Device.

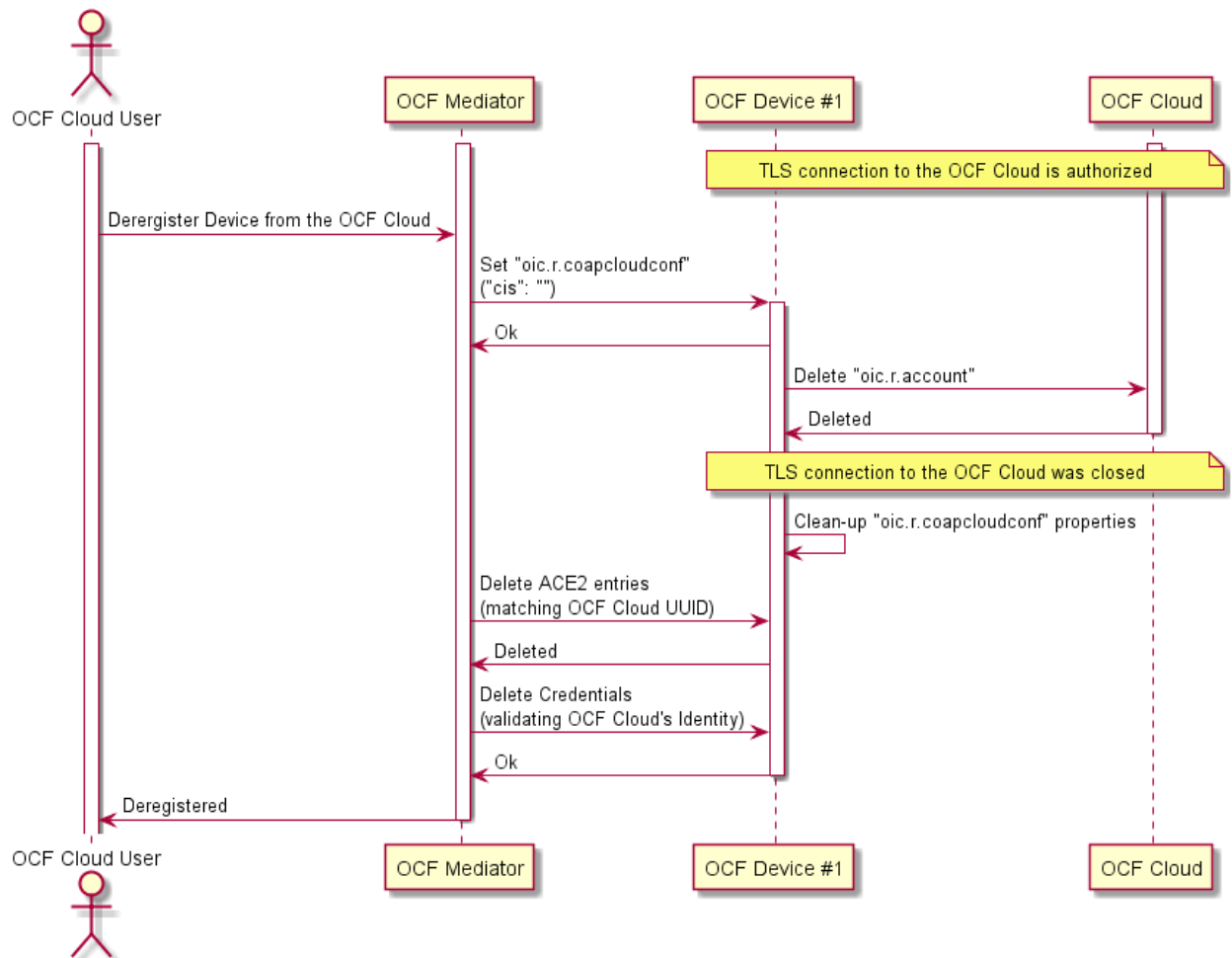


Figure 3 – Device deregistration from a Mediator flow

420
421

7 Device authentication with OCF Cloud

7.1 Device authentication with OCF Cloud general

424 The mechanisms for Device Authentication in clauses 10.2, 10.3 and 10.4 of ISO/IEC 30118-2
 425 imply that a Device is authorized to communicate with any other Device meeting the criteria
 426 provisioned in "/oic/sec/cred"; the "/oic/sec/acl2" Resource (or "/oic/sec/acl1" Resource of OIC1.1
 427 Servers) are additionally used to restrict access to specific Resources. The present clause
 428 describes Device authentication for OCF Cloud, which uses slightly different criteria as described
 429 in ISO/IEC 30118-2. A Device accessing an OCF Cloud shall establish a TLS session. The mutual
 430 authenticated TLS session is established using Server certificate and Client certificate.

431 Each Device is identified by the Access Token obtained from the Device Registration response.
 432 The OCF Cloud holds an OCF Cloud association table that maps Access Token, User ID and Device
 433 UUID. The Device Registration shall happen while the Device is in RFNOP state. After Device
 434 Registration, the updated Access Token, Device UUID and User ID are used by the Device for the
 435 subsequent connection with the OCF Cloud.

7.2 Device connection with the OCF Cloud

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

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

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

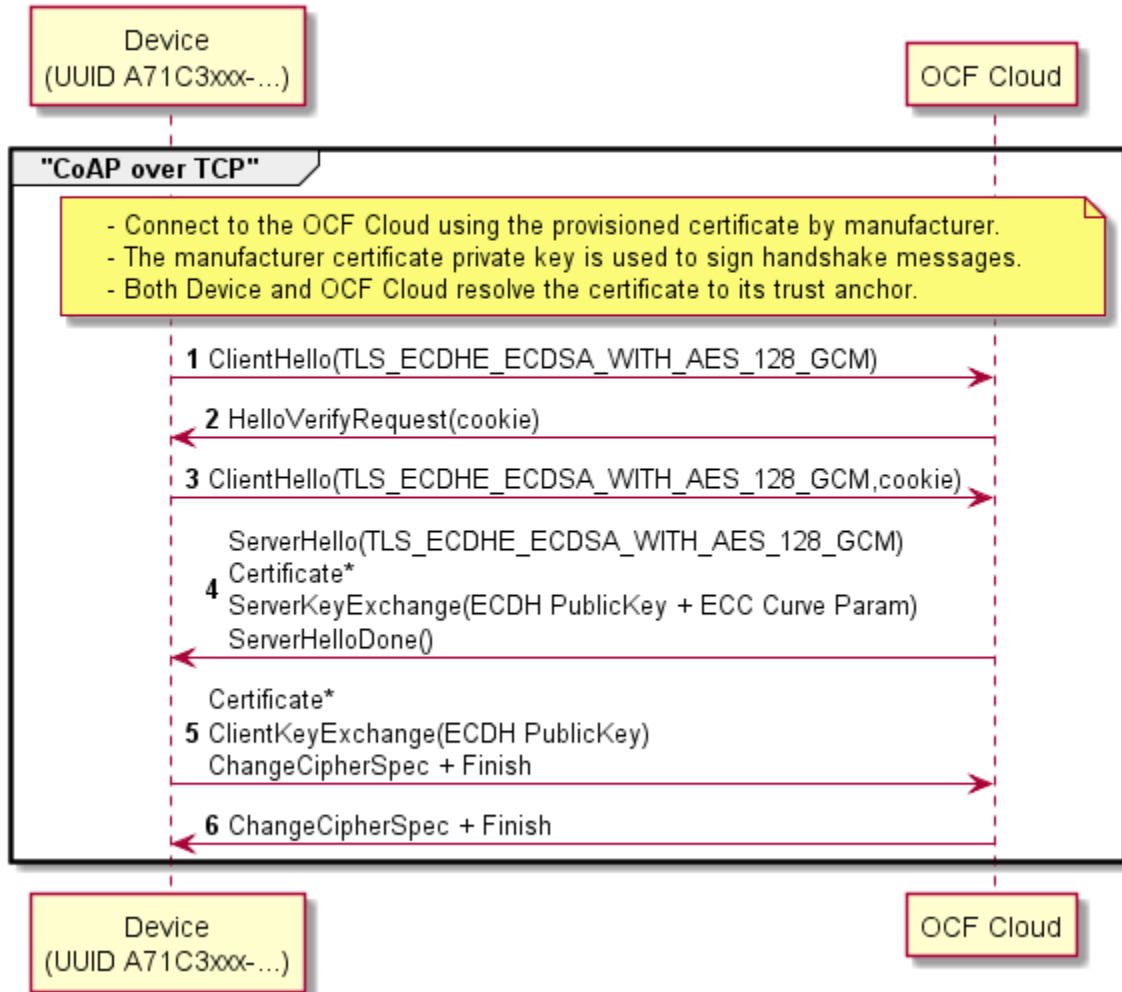
444 The OCF Cloud is expected to validate the manufacturer certificate provided by the Device.

445 The assumption is that the OCF Cloud User trusts the OCF Cloud that the Device connects. The
446 OCF Cloud connection should not happen without the consent of the OCF Cloud User. The
447 assumption is that the OCF Cloud User has either service agreement with the OCF Cloud provider
448 or uses manufacturer provided OCF Cloud.

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

453 Figure 4 depicts sequence for Device connection with OCF Cloud and steps described in Table 2.
454 .

Device Connection with OCF Cloud



455

456

Figure 4 – Device connection with OCF Cloud

457

458

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

459

7.3 Security considerations

460

461

462

463

464

465

466

467

468

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

469 The security considerations about the manufacturer certificate as described in clause 7.3.6.5 of
470 ISO/IEC 30118-2 are also applicable in the Device authentication with the OCF Cloud.

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

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

476 **8 Message integrity and confidentiality**

477 **8.1 OCF Cloud session semantics**

478 The messages between the OCF Cloud and Device shall be exchanged only if the Device and OCF
479 Cloud authenticate each other as described in 6.3. The asymmetric cipher suites as described in
480 8.2 shall be employed for establishing a secured session and for encrypting/decrypting between
481 the OCF Cloud and the Device. The OCF Endpoint sending the message shall encrypt and
482 authenticate the message using the cipher suite as described in 8.2 and the OCF Endpoint shall
483 verify and decrypt the message before processing it.

484 **8.2 Cipher suites for OCF Cloud Credentials**

485 All Devices supporting OCF Cloud Certificate Credentials shall implement:

486 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256

487 All Devices supporting OCF Cloud Certificate Credentials should implement:

488 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,

489 TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256,

490 TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,

491 TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,

492 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

493

494 **9 Security Resources**

495 **9.1 Account Resource**

496 The Account Resource specifies the Properties based on ISO/IEC 17788 *Information technology –*
497 *Cloud computing – Overview and vocabulary*

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

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

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

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

506 During the Device Registration process, an OCF Cloud can provide a distinct URI of another OCF
507 Cloud ("redirected-to" OCF Cloud). Both initial and redirected-to OCF Clouds are expected to
508 belong to the same Vendor; they are assumed to have the same UUID and are assumed to have
509 an Out-of-Band Communication Channel established. Device does not have to perform the Device
510 Registration on the redirected-to OCF Cloud and the OCF Cloud may ignore such attempts.

511 Redirected-to OCF Cloud is expected to accept the Access Token, provided to the Device by the
512 initial OCF Cloud.

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

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

516 – A Device intending to register with the OCF Cloud shall send UPDATE with following Properties
517 "di" ("di" Property Value of "/oic/d" Resource), and "accesstoken" as configured by the Mediator
518 ("at" Property Value of "oic.r.coapcloudconf" Resource). The OCF Cloud verifies it is the same
519 "accesstoken" which was assigned to the Mediator for the corresponding "di" Property Value.
520 The "accesstoken" is the permission for the Device to access the OCF Cloud. If the "apn" was
521 included when the Mediator UPDATED the "oic.r.coapcloudconf" Resource, the Device shall
522 also include "authprovider" Property when registering with the OCF Cloud. If no "apn" is
523 specified, then the "authprovider" Property shall not be included in the UPDATE request.

524 OCF Cloud returns "accesstoken", "uid", "refreshtoken", and "expiresin" It may also return
525 "redirecturi". Received "accesstoken" is to be treated by Device as an Access Token with "Bearer"
526 token type as defined in IETF RFC 6750. This "accesstoken" shall be used for the following Account
527 Session start using "oic/sec/session" SVR. Received "refreshtoken" is to be treated by Device as
528 a Refresh Token as defined in ISO/IEC 17788 *Information technology – Cloud computing –*
529 *Overview and vocabulary*
530 <https://www.iso.org/standard/60544.html>

531 ISO/IEC 17789 *Information technology – Cloud computing – Reference architecture*
532 <https://www.iso.org/standard/60545.html>

533 – IETF RFC 6749. The Device stores the OCF Cloud's Response values. If "redirecturi" is
534 received, Device shall use received value as a new OCF Cloud URI instead of "cis" Property
535 Value of "oic.r.coapcloudconf" Resource for further connections.

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

537 – To deregister with the OCF Cloud, a DELETE operation shall be sent. If the session has not
538 been created and the TLS connection is not authorized, DELETE operation shall be sent with
539 the "accesstoken" and either the "uid" or "di" Properties to be deregistered with the OCF Cloud
540 as query parameters. In case the "di" Property is omitted in a DELETE operation, the OCF
541 Cloud is expected to deregister the Device with a matching "accesstoken" Property value. In
542 case the session is already created and the TLS connection is already authorized, no query
543 parameters ("accesstoken", "uid", or "di") are required in the DELETE request. On DELETE with
544 the OCF Cloud, the Device should also delete values internally stored. Once deregistered from
545 an OCF Cloud, a Device can connect to any other OCF Cloud. Device deregistered needs to
546 go through the steps in clause 6 again to be registered with the OCF Cloud.

547 The "oic.r.account" Resource is defined in Table 3. Complete details are provided in annex A.2.

548 **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.baseline	Resource used for a Device to add itself under a given credential	N/A

549 Table 4 defines the Properties of the "oic.r.account" Resource Type.

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

Property Title	Property Name	Value Type	Value Rule	Access Mode	Mandatory	Description
Device UUID	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 UUID, or the Authorization Code which is then verified and exchanged for the Access Token during Device Registration. Property is not required if the TLS connection is already authorized.
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 UUID.
Refresh Token	refreshtoken	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.

551 9.2 Account Session Resource

552 The "/oic/sec/session" Resource hosted on the OCF Cloud is used for creating connections with
 553 the OCF Cloud subsequent to Device registration though "/oic/sec/account" Resource. The
 554 "/oic/sec/session" Resource requires the Device UUID, User ID and Access Token which are stored
 555 securely on the Device.

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

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

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

- 561 – The Device connecting to the OCF Cloud shall send an UPDATE request message to the OCF
 562 Cloud's "/oic/sec/session" Resource. The message shall include the "di" Property Value of
 563 "/oic/d" Resource and "uid", "login" Value ("true" to establish connection; "false" to disconnect)
 564 and "accesstoken" as returned by OCF Cloud during Device Registration. The OCF Cloud
 565 verifies it is the same Access Token which was returned to the Device during Device
 566 Registration process or during Token Refresh. If Device was attempting to establish the
 567 connection and provided values were verified as correct by the OCF Cloud, OCF Cloud sends
 568 a response with remaining lifetime of the associated Access Token ("expiresin" Property Value).

569 The "oic.r.session" Resource is defined in Table 5.

570

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

571 Table 6 defines the Properties of the "oic.r.session" Resource. Complete details are provided in
572 annex A.3.

573

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 UUID	di	string	uuid	W	Yes	Unique Device UUID 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 UUID
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

574 9.3 Account Token Refresh Resource

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

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

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

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

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

583 – The Device attempting to refresh the Access Token shall send an UPDATE request message
584 to the OCF Cloud's "/oic/sec/tokenrefresh" Resource. The message shall include the "di"
585 Property Value of "/oic/d" Resource, "uid" and "refreshtoken", as returned by OCF Cloud.

586 OCF Cloud response is expected to include a "refreshtoken", new "accesstoken", and "expiresin".
587 Received "accesstoken" is to be treated by Device as an Access Token with "Bearer" token type
588 as defined in IETF RFC 6750. This Access Token is the permission for the Device to access the
589 OCF Cloud. Received "refreshtoken" is to be treated by Device as a Refresh Token as defined in
590 ISO/IEC 17788 *Information technology – Cloud computing – Overview and vocabulary*
591 <https://www.iso.org/standard/60544.html>

592 ISO/IEC 17789 *Information technology – Cloud computing – Reference architecture*
 593 <https://www.iso.org/standard/60545.html>
 594 – IETF RFC 6749. Received "refreshtoken" may be the new Refresh Token or the same one as
 595 provided by the Device in the UPDATE request. In case when new distinct "refreshtoken" is
 596 provided by the OCF Cloud, the Device shall discard the old value. The OCF Cloud's response
 597 values "refreshtoken", "acesstoken" and "expiresin" are securely stored on the Device.
 598 The "oic.r.tokenrefresh" Resource is defined in Table 7. Complete details are provided in annex
 599 A.4.

600 **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

601 Table 8 defines the Properties of the "oic.r.tokenrefresh" Resource.

602 **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 UUID	di	string	uuid	W	Yes	Unique Device UUID 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 UUID.
Token Expiration	expiresin	integer	-	R	Yes	Access Token life time in seconds (-1 if permanent).

603 **10 Security hardening guidelines**

604 **10.1 Security hardening guidelines general**

605 In addition to the Sensitive Data list outlined in Table 75 of ISO/IEC 30118-2, any Device
 606 implementing OCF Cloud connection capabilities should also provide reasonable protection for the
 607 information in Table 9.

608 **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

609

Annex A (normative) Resource Type definitions

610
611
612

613 A.1 List of Resource Type definitions

614 All the clauses in Annex A describe the Resource Types with a RESTful API definition language.
615 The Resource Type definitions presented in Annex A are formatted for readability, and so may
616 appear to have extra line breaks.

617 Table A.1 contains the list of defined security Resources in this document.

618 **Table A.1 – Alphabetized list of security Resources**

Friendly Name (informative)	Resource Type (rt)	Clause
Account	oic.r.account	A.2
Account Session	oic.r.session	A.3
Account Token Refresh	oic.r.tokenrefresh	A.4

619 A.2 Account Token

620 A.2.1 Introduction

621 Sign-up using generic account provider.

622 A.2.2 Well-known URI

623 /oic/sec/account

624 A.2.3 Resource type

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

626 A.2.4 OpenAPI 2.0 definition

```
627 {  
628   "swagger": "2.0",  
629   "info": {  
630     "title": "Account Token",  
631     "version": "20190111",  
632     "license": {  
633       "name": "OCF Data Model License",  
634       "url":  
635       "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI  
636       CENSE.md",  
637       "x-copyright": "copyright 2016-2017, 2019 Open Connectivity Foundation, Inc. All rights  
638       reserved."  
639     },  
640     "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"  
641   },  
642   "schemes": ["http"],  
643   "consumes": ["application/json"],  
644   "produces": ["application/json"],  
645   "paths": {  
646     "/oic/sec/account" : {  
647       "post": {  
648         "description": "Sign-up using generic account provider.\n",  
649         "parameters": [  
650           {"$ref": "#/parameters/interface"},  
651           {  
652             "name": "body",  
653             "in": "body",  
654             "required": true,  
655             "schema": { "$ref": "#/definitions/Account-request" },  
656             "x-example":  
657             {
```

```

658         "di" : "9cfbeb8e-5a1e-4d1c-9d01-00c04fd430c8",
659         "authprovider" : "github",
660         "accesstoken" : "8802f2eaf8b5e147a936"
661     }
662 }
663 ],
664 "responses": {
665     "204": {
666         "description" : "2.04 Changed respond with required and optional information\n",
667         "x-example":
668         {
669             "rt": ["oic.r.account"],
670             "accesstoken" : "0f3d9f7fe5491d54077d",
671             "refresh token" : "00fe4644a6fbe5324eec",
672             "expiresin" : 3600,
673             "uid" : "123e4567-e89b-12d3-a456-d6e313b71d9f",
674             "redirecturi" : "coaps+tcp://example.com:443"
675         },
676         "schema": { "$ref": "#/definitions/Account-response" }
677     }
678 }
679 },
680 "delete": {
681     "description": "Delete a device. This also removes all resources in the device on cloud
682 side.\nexample: /oic/account?di=9cfbeb8e-5a1e-4d1c-9d01-
683 00c04fd430c8&accesstoken=0f3d9f7fe5491d54077d\n",
684     "parameters": [
685         {"$ref": "#/parameters/interface"}
686     ],
687     "responses": {
688         "202": {
689             "description" : "2.02 Deleted response informing the device is successfully
690 deleted.\n"
691         }
692     }
693 }
694 }
695 },
696 "parameters": {
697     "interface" : {
698         "in" : "query",
699         "name" : "if",
700         "type" : "string",
701         "enum" : ["oic.if.baseline"]
702     }
703 },
704 "definitions": {
705     "Account-request" : {
706         "properties": {
707             "authprovider": {
708                 "description": "The name of Authorization Provider through which Access Token was
709 obtained",
710                 "type": "string"
711             },
712             "accesstoken" : {
713                 "description": "Access-Token used for communication with OCF Cloud after account
714 creation",
715                 "pattern": "(?!$|\\s+).*",
716                 "type": "string"
717             },
718             "di": {
719                 "description": "Format pattern according to IETF RFC 4122.",
720                 "pattern": "^([a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
721 9]{12})$",
722                 "type": "string"
723             }
724         },
725         "type" : "object",
726         "required": ["di", "accesstoken"]
727     },
728     "Account-response": {
729         "properties": {

```

```

730     "expiresin" : {
731         "description": "Access-Token remaining life time in seconds (-1 if permanent)",
732         "readOnly": true,
733         "type": "integer"
734     },
735     "rt": {
736         "description": "Resource Type of the Resource",
737         "items": {
738             "maxLength": 64,
739             "type": "string",
740             "enum" : ["oic.r.account"]
741         },
742         "minItems": 1,
743         "maxItems": 1,
744         "readOnly": true,
745         "type": "array"
746     },
747     "refreshToken" : {
748         "description": "Refresh token can be used to refresh the Access Token before getting
749 expired",
750         "pattern": "(?!$|\\s+).*",
751         "readOnly": true,
752         "type": "string"
753     },
754     "uid" : {
755         "description": "Format pattern according to IETF RFC 4122.",
756         "pattern": "^-[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
757 9]{12}$",
758         "type": "string"
759     },
760     "accesstoken" : {
761         "description": "Access-Token used for communication with cloud after account creation",
762         "pattern": "(?!$|\\s+).*",
763         "type": "string"
764     },
765     "n": {
766         "$ref":
767 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
768 schema.json#/definitions/n"
769     },
770     "id": {
771         "$ref":
772 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
773 schema.json#/definitions/id"
774     },
775     "redirecturi" : {
776         "description": "Using this URI, the Client needs to reconnect to a redirected OCF Cloud.
777 If provided, this value shall be used by the Device instead of Mediator-provided URI during the
778 Device Registration.",
779         "readOnly": true,
780         "type": "string"
781     },
782     "if": {
783         "description": "The interface set supported by this resource",
784         "items": {
785             "enum": [
786                 "oic.if.baseline"
787             ],
788             "type": "string"
789         },
790         "minItems": 1,
791         "maxItems": 1,
792         "uniqueItems": true,
793         "readOnly": true,
794         "type": "array"
795     }
796 },
797 "type" : "object",
798 "required": ["accesstoken", "refreshToken", "expiresin", "uid"]
799 }
800 }

```

801 }
802

803 **A.2.5 Property definition**

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

805 **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 UUID, 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	
refresh token	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 UUID.
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	

806 **A.2.6 CRUDN behaviour**

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

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

Create	Read	Update	Delete	Notify
		post	delete	

809 **A.3 Session**

810 **A.3.1 Introduction**

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

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

813 /oic/sec/session

814 **A.3.3 Resource type**

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

816 **A.3.4 OpenAPI 2.0 definition**

```

817 {
818   "swagger": "2.0",
819   "info": {
820     "title": "Session",
821     "version": "v1.0-20181001",
822     "license": {
823       "name": "OCF Data Model License",
824       "url":
825 "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI
826 CENSE.md",
827       "x-copyright": "copyright 2016-2017, 2019 Open Connectivity Foundation, Inc. All rights
828 reserved."
829     },

```

```

830     "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
831   },
832   "schemes": ["http"],
833   "consumes": ["application/json"],
834   "produces": ["application/json"],
835   "paths": {
836     "/oic/sec/session" : {
837       "post": {
838         "description": "Resource that manages the persistent session between a Device and OCF
839 Cloud.",
840         "parameters": [
841           {"$ref": "#/parameters/interface"},
842           {
843             "name": "body",
844             "in": "body",
845             "required": true,
846             "schema": { "$ref": "#/definitions/Account-Session-Request" },
847             "x-example":
848               {
849                 "uid" : "123e4567-e89b-12d3-a456-d6e313b71d9f",
850                 "di" : "9cfbeb8e-5ale-4d1c-9d01-00c04fd430c8",
851                 "accesstoken" : "0f3d9f7fe5491d54077d",
852                 "login" : true
853               }
854           }
855         ],
856         "responses": {
857           "204": {
858             "description": "",
859             "x-example":
860               {
861                 "rt": ["oic.r.session"],
862                 "expiresin" : 3600
863               },
864             "schema": { "$ref": "#/definitions/Account-Session-Response" }
865           }
866         }
867       }
868     }
869   },
870   "parameters": {
871     "interface" : {
872       "in" : "query",
873       "name" : "if",
874       "type" : "string",
875       "enum" : ["oic.if.baseline"]
876     }
877   },
878   "definitions": {
879     "Account-Session-Request" : {
880       "properties": {
881         "uid": {
882           "description": "Format pattern according to IETF RFC 4122.",
883           "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
884 9]{12}$",
885           "type": "string"
886         },
887         "di": {
888           "description": "The Device UUID\nFormat pattern according to IETF RFC 4122.",
889           "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
890 9]{12}$",
891           "type": "string"
892         },
893         "accesstoken": {
894           "description": "Access-Token used to grant access right for the Device to sign-in.",
895           "pattern": "(?!$|\\s+).*",
896           "type": "string"
897         },
898         "login": {
899           "description": "Action for the request: true = login, false = logout.",
900           "type": "boolean"
901         }

```

```

902     },
903     "type" : "object",
904     "required": ["uid", "di", "accesstoken", "login"]
905 },
906 "Account-Session-Response" : {
907     "properties": {
908         "expiresin": {
909             "description": "Access-Token remaining life time in seconds (-1 if permanent).",
910             "readOnly": true,
911             "type": "integer"
912         },
913         "rt": {
914             "description": "Resource Type of the Resource.",
915             "items": {
916                 "maxLength": 64,
917                 "type": "string",
918                 "enum": ["oic.r.session"]
919             },
920             "minItems": 1,
921             "readOnly": true,
922             "type": "array"
923         },
924         "n": {
925             "$ref":
926 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
927 schema.json#/definitions/n"
928         },
929         "id": {
930             "$ref":
931 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
932 schema.json#/definitions/id"
933         },
934         "if": {
935             "description": "The interface set supported by this Resource.",
936             "items": {
937                 "enum": [
938                     "oic.if.baseline"
939                 ],
940                 "type": "string"
941             },
942             "minItems": 1,
943             "readOnly": true,
944             "type": "array"
945         }
946     },
947     "type" : "object",
948     "required" : ["expiresin"]
949 }
950 }
951 }
952

```

953 A.3.5 Property definition

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

955 **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 UUID 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 UUID.
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.

956 **A.3.6 CRUDN behaviour**

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

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

Create	Read	Update	Delete	Notify
		post		

959 **A.4 Token Refresh**

960 **A.4.1 Introduction**

961 Obtain fresh Access Token using the refresh token, client should refresh Access Token before it
962 expires.

963 **A.4.2 Well-known URI**

964 /oic/sec/tokenrefresh

965 A.4.3 Resource type

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

967 A.4.4 OpenAPI 2.0 definition

```
968 {
969   "swagger": "2.0",
970   "info": {
971     "title": "Token Refresh",
972     "version": "v1.0-20181001",
973     "license": {
974       "name": "OCF Data Model License",
975       "url":
976 "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI
977 CENSE.md",
978     "x-copyright": "copyright 2016-2017, 2019 Open Connectivity Foundation, Inc. All rights
979 reserved."
980   },
981   "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
982 },
983   "schemes": ["http"],
984   "consumes": ["application/json"],
985   "produces": ["application/json"],
986   "paths": {
987     "/oic/sec/tokenrefresh" : {
988       "post": {
989         "description": "Obtain fresh access-token using the refresh token, client should refresh
990 access-token before it expires.\n",
991         "parameters": [
992           {"$ref": "#/parameters/interface"},
993           {
994             "name": "body",
995             "in": "body",
996             "required": true,
997             "schema": { "$ref": "#/definitions/TokenRefresh-Request" },
998             "x-example":
999             {
1000               "uid" : "123e4567-e89b-12d3-a456-d6e313b71d9f",
1001               "di" : "9cfbeb8e-5ale-4dlc-9d01-00c04fd430c8",
1002               "refreshToken" : "00fe4644a6fbe5324eec"
1003             }
1004           }
1005         ],
1006         "responses": {
1007           "204": {
1008             "description": "2.04 Changed respond with new access-token.\n",
1009             "x-example":
1010             {
1011               "rt": ["oic.r.tokenrefresh"],
1012               "accessToken" : "8ce598980761869837be",
1013               "refreshToken" : "d4922312b6df0518e146",
1014               "expiresin" : 3600
1015             }
1016           },
1017           "schema": { "$ref": "#/definitions/TokenRefresh-Response" }
1018         }
1019       }
1020     }
1021   },
1022   "parameters": {
1023     "interface" : {
1024       "in" : "query",
1025       "name" : "if",
1026       "type" : "string",
1027       "enum" : ["oic.if.baseline"]
1028     }
1029   },
1030   "definitions": {
1031     "TokenRefresh-Request" : {
1032       "properties": {
1033         "refreshToken": {
1034
```

```

1035         "description": "Refresh token received by account management or during token refresh
1036 procedure.",
1037         "pattern": "(?!$|\\s+).*",
1038         "type": "string"
1039     },
1040     "uid": {
1041         "description": "Format pattern according to IETF RFC 4122.",
1042         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
1043 9]{12}$",
1044         "type": "string"
1045     },
1046     "di": {
1047         "description": "Format pattern according to IETF RFC 4122.",
1048         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
1049 9]{12}$",
1050         "type": "string"
1051     }
1052 },
1053 "type" : "object",
1054 "required": ["uid", "di", "refreshtoken"]
1055 },
1056 "TokenRefresh-Response" : {
1057     "properties": {
1058         "expiresin": {
1059             "description": "Access-Token life time in seconds (-1 if permanent).",
1060             "readOnly": true,
1061             "type": "integer"
1062         },
1063         "rt": {
1064             "description": "Resource Type of the Resource.",
1065             "items": {
1066                 "maxLength": 64,
1067                 "type": "string",
1068                 "enum": ["oic.r.tokenrefresh"]
1069             },
1070             "minItems": 1,
1071             "readOnly": true,
1072             "type": "array"
1073         },
1074         "refreshtoken": {
1075             "description": "Refresh token received by account management or during token refresh
1076 procedure.",
1077             "pattern": "(?!$|\\s+).*",
1078             "type": "string"
1079         },
1080         "accesstoken": {
1081             "description": "Granted Access-Token.",
1082             "pattern": "(?!$|\\s+).*",
1083             "readOnly": true,
1084             "type": "string"
1085         },
1086         "n": {
1087             "$ref":
1088 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1089 schema.json#/definitions/n"
1090         },
1091         "id": {
1092             "$ref":
1093 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1094 schema.json#/definitions/id"
1095         },
1096         "if" :
1097         {
1098             "description": "The interface set supported by this Resource.",
1099             "items": {
1100                 "enum": [
1101                     "oic.if.baseline"
1102                 ],
1103                 "type": "string"
1104             },
1105             "minItems": 1,
1106             "readOnly": true,

```

```

1107     "type": "array"
1108   },
1109 },
1110 "type" : "object",
1111 "required": ["accesstoken", "refreshtoken", "expiresin"]
1112 }
1113 }
1114 }
1115

```

1116 A.4.5 Property definition

1117 Table A.6 defines the Properties that are part of the "oic.r.tokenrefresh" Resource Type.

1118 **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 UUID 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 UUID.
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	

1119 **A.4.6 CRUDN behaviour**

1120 Table A.7 defines the CRUDN operations that are supported on the "oic.r.tokenrefresh" Resource
 1121 Type.

1122 **Table A.7 – The CRUDN operations of the Resource with type "rt" = "oic.r.tokenrefresh".**

Create	Read	Update	Delete	Notify
		post		

1123

1124