

1                   **OCF “Essen” – Cloud Security Specification – Security WG CR 2871**

2  
3  
4                   Legal Disclaimer

5  
6 THIS IS A DRAFT SPECIFICATION DOCUMENT ONLY AND HAS NOT BEEN ADOPTED BY THE  
7 OPEN CONNECTIVITY FOUNDATION. THIS DRAFT DOCUMENT MAY NOT BE RELIED UPON  
8 FOR ANY PURPOSE OTHER THAN REVIEW OF THE CURRENT STATE OF THE DEVELOPMENT  
9 OF THIS DRAFT DOCUMENT. THE OPEN CONNECTIVITY FOUNDATION AND ITS MEMBERS  
10 RESERVE THE RIGHT WITHOUT NOTICE TO YOU TO CHANGE ANY OR ALL PORTIONS  
11 HEREOF, DELETE PORTIONS HEREOF, MAKE ADDITIONS HERETO, DISCARD THIS DRAFT  
12 DOCUMENT IN ITS ENTIRETY OR OTHERWISE MODIFY THIS DRAFT DOCUMENT AT ANY  
13 TIME. YOU SHOULD NOT AND MAY NOT RELY UPON THIS DRAFT DOCUMENT IN ANY WAY,  
14 INCLUDING BUT NOT LIMITED TO THE DEVELOPMENT OF ANY PRODUCTS OR SERVICES.  
15 IMPLEMENTATION OF THIS DRAFT DOCUMENT IS DONE AT YOUR OWN RISK AMEND AND  
16 IT IS NOT SUBJECT TO ANY LICENSING GRANTS OR COMMITMENTS UNDER THE OPEN  
17 CONNECTIVITY FOUNDATION INTELLECTUAL PROPERTY RIGHTS POLICY OR OTHERWISE.  
18 IN CONSIDERATION OF THE OPEN CONNECTIVITY FOUNDATION GRANTING YOU ACCESS  
19 TO THIS DRAFT DOCUMENT, YOU DO HEREBY WAIVE ANY AND ALL CLAIMS ASSOCIATED  
20 HERewith INCLUDING BUT NOT LIMITED TO THOSE CLAIMS DISCUSSED BELOW, AS WELL  
21 AS CLAIMS OF DETRIMENTAL RELIANCE.

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

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

25 Copying or other form of reproduction and/or distribution of these works are strictly prohibited.  
26

# OCF Cloud Security Specification

VERSION 1 | June 20, 2019



27 **LEGAL DISCLAIMER**

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

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

41 Copyright © 2017-2019 Open Connectivity Foundation, Inc. All rights reserved.

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

DRAFT

43	1	Purpose and Role.....	1
44	2	Normative References .....	1
45	3	Terms, definitions, and abbreviated terms .....	2
46	3.1	Terms and definitions.....	2
47	3.2	Abbreviated terms.....	3
48	4	Document Conventions and Organization .....	3
49	4.1	Conventions.....	3
50	4.2	Notation.....	3
51	4.3	Data types .....	4
52	4.4	Document structure.....	4
53	5	Security overview .....	6
54	5.1	Preamble .....	6
55	5.2	Device Provisioning for OCF Cloud and Device Registration Overview.....	6
56	5.3	Credential overview .....	6
57	6	Device provisioning for OCF Cloud.....	6
58	6.1	Cloud Provisioning General .....	6
59	6.2	Device Provisioning by Mediator .....	7
60	7	Device authentication with OCF Cloud.....	10
61	7.1	Device Authentication with OCF Cloud General .....	10
62	7.2	Device Connection with the OCF Cloud .....	10
63	7.3	Security Considerations .....	12
64	8	Message integrity and confidentiality .....	13
65	8.1	Cloud Session Semantics .....	13
66	8.2	Cipher suites for OCF Cloud Credentials .....	13
67	9	Security resources.....	13
68	9.1	Account Resource.....	13
69	9.2	Account Session resource.....	15
70	9.3	Account Token Refresh Resource .....	16
71	10	Security hardening guidelines.....	17
72		Annex A (normative) Resource Type definitions .....	19
73	A.1	Account Token.....	19
74	A.1.1	Introduction .....	19
75	A.1.2	Well-known URI.....	19
76	A.1.3	Resource type .....	19
77	A.1.4	OpenAPI 2.0 definition.....	19
78	A.1.5	Property definition .....	22
79	A.1.6	CRUDN behaviour .....	23
80	A.2	Session.....	23
81	A.2.1	Introduction .....	23
82	A.2.2	Well-known URI.....	24



83	A.2.3	Resource type .....	24
84	A.2.4	OpenAPI 2.0 definition.....	24
85	A.2.5	Property definition .....	26
86	A.2.6	CRUDN behaviour .....	27
87	A.3	Token Refresh .....	27
88	A.3.1	Introduction .....	27
89	A.3.2	Well-known URI .....	27
90	A.3.3	Resource type .....	27
91	A.3.4	OpenAPI 2.0 definition.....	27
92	A.3.5	Property definition .....	29
93	A.3.6	CRUDN behaviour .....	30
94			

DRAFT

95	<b>FIGURES</b>	
96	Figure 1 – OCF Interaction.....	3
97	Figure 2 – Device connection with OCF Cloud .....	12
98		
99		

100	<b>Tables</b>	
101	Table 1 – Mapping of Properties of the "oic.r.account" and "oic.r.coapcloudconf"	
102	Resources .....	10
103	Table 2 – Device connection with the OCF Cloud flow .....	12
104	Table 3 – Definition of the "oic.r.account" Resource .....	14
105	Table 4 – Properties of the "oic.r.account" Resource .....	15
106	Table 5 – Definition of the "oic.r.session" Resource .....	16
107	Table 6 – Properties of the "oic.r.session" Resource .....	16
108	Table 7 – Definition of the "oic.r.tokenrefresh" Resource .....	17
109	Table 8 – Properties of the "oic.r.tokenrefresh" Resource .....	17
110	Table 9 – Sensitive Data related to OCF Cloud .....	18
111	Table A.1 – Alphabetized list of security resources .....	19
112	Table A.2 – The Property definitions of the Resource with type "rt" = "oic.r.account". .....	22
113	Table A.3 – The CRUDN operations of the Resource with type "rt" = "oic.r.account". .....	23
114	Table A.4 – The Property definitions of the Resource with type "rt" = "oic.r.session". .....	26
115	Table A.5 – The CRUDN operations of the Resource with type "rt" = "oic.r.session". .....	27
116	Table A.6 – The Property definitions of the Resource with type "rt" = "oic.r.tokenrefresh". .....	29
117	Table A.7 – The CRUDN operations of the Resource with type "rt" = "oic.r.tokenrefresh". .....	31
118		

119 **1 Purpose and Role**

120 This document defines security objectives, philosophy, resources and mechanism that impacts  
121 OCF base layers of ISO/IEC 30118-1:2018. ISO/IEC 30118-1:2018 contains informative security  
122 content. The OCF Security Specification contains security normative content and may contain  
123 informative content related to the OCF base or other OCF documents.

124 **2 Normative References**

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

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

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

135 OCF Security Specification, Information technology – Open Connectivity Foundation (OCF)  
136 Specification, Latest version available  
137 at:[https://openconnectivity.org/specs/OCF\\_Security\\_Specification.pdf](https://openconnectivity.org/specs/OCF_Security_Specification.pdf)

138 OCF Cloud Specification, Information technology – Open Connectivity Foundation (OCF)  
139 Specification – Part 8: Cloud Specification, Latest version available at:  
140 [https://openconnectivity.org/specs/OCF\\_Cloud\\_Specification.pdf](https://openconnectivity.org/specs/OCF_Cloud_Specification.pdf)

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

143 IETF RFC 6750, *The OAuth 2.0 Authorization Framework: Bearer Token Usage*, October 2012,  
144 <https://tools.ietf.org/html/rfc6750>  
145

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

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

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

151

152

153

154

155

156

## 157 **3 Terms, definitions, and abbreviated terms**

### 158 **3.1 Terms and definitions**

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

161 ISO and IEC maintain terminological databases for use in standardization at the following  
162 addresses:

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

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

#### 165 **3.1.1**

##### 166 **Access Management Service (AMS)**

167 dynamically constructs ACL Resources in response to a Device Resource request.

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

#### 170 **3.1.2**

##### 171 **Trust Anchor**

172 a well-defined, shared authority, within a trust hierarchy, by which two cryptographic entities (e.g.  
173 a Device and an onboarding tool) can assume trust

#### 174 **3.1.3**

##### 175 **OCF Security Domain**

176 set of onboarded OCF Devices that are provisioned with credentialing information for confidential  
177 communication with one another

#### 178 **3.1.4**

##### 179 **Access Token**

180 a credential used to authorize the connection with the OCF Cloud and access protected resources.  
181 An Access Token is a string while the OCF Device has no internal logic based on its contents and  
182 only forwards the token as-is

#### 183 **3.1.5**

##### 184 **Authorization Provider**

185 a Server issuing Access Tokens (3.1.4) to the Client after successfully authenticating the OCF  
186 Cloud User (3.1.7) and obtaining authorization.

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

#### 188 **3.1.6**

##### 189 **Device Registration**

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

#### 193 **3.1.7**

##### 194 **OCF Cloud User**

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

196 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  
197 organization that onboarded that Device. The OCF Cloud User delegates, to the OCF Cloud authority, authority to route



198 between Devices registered by the OCF Cloud User. The OCF Cloud delegates, to the OCF Cloud User, authority to  
199 select the set of Devices which can register and use the services of the OCF Cloud.

200 **3.2 Abbreviated terms**

201 **3.2.1**

202 **ACE**

203 Access Control Entry

204 **3.2.2**

205 **ACL**

206 Access Control List

207 **3.2.3**

208 **AMS**

209 Access Management Service

210 **3.2.4**

211 **CMS**

212 Credential Management Service

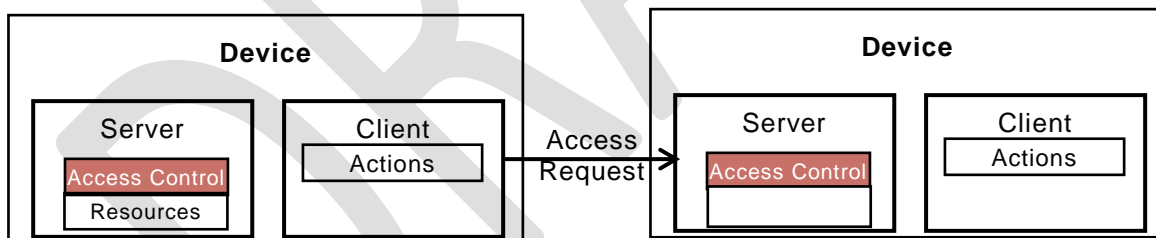
213 **4 Document Conventions and Organization**

214 **4.1 Conventions**

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

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

218 Figure 1 depicts interaction between OCF Devices.



219

220

**Figure 1 – OCF Interaction**

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

225 **4.2 Notation**

226 In this document, features are described as required, recommended, allowed or DEPRECATED as  
227 follows:

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

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

232 **Recommended (or should).**

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

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

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

242 **Conditionally allowed (CA)**

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

245 **Conditionally required (CR)**

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

249 **DEPRECATED**

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

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

257 Words that are emphasized are printed in italic.

258 **4.3 Data types**

259 See ISO/IEC 30118-1:2018.

260 **4.4 Document structure**

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

263 The Security Specification may use the oneM2M Release 3 Specifications,  
264 <http://www.onem2m.org/technical/published-drafts>

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

DRAFT

## 268 **5 Security overview**

### 269 **5.1 Preamble**

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

- 272 – Device and Mediator connect over DTLS using "/oic/sec/cred"
- 273 – Device is provisioned by Mediator with following information:
  - 274 – the URL of OCF Cloud
  - 275 – Authorization Provider Name to identify the origin of the Access Token
  - 276 – Access Token / Authorization Code that is validated / exchanged by the OCF Cloud
  - 277 – UUID of the OCF Cloud

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

### 281 **5.2 Device Provisioning for OCF Cloud and Device Registration Overview**

282 As mentioned in the start of Clause 0, communication between a Device and OCF Cloud is subject  
283 to different criteria in comparison to Devices which are within a single local network. The Device is  
284 configured in order to connect to the OCF Cloud by a Mediator as specified in the CoAPCloudConf  
285 Resource clauses in OCF Cloud Specification. Provisioning includes the remote connectivity and  
286 local details such as URL where the OCF Cloud hosting environment can be found, the OCF Cloud  
287 verifiable Access Token and optionally the name of the Authorization Provider which issued the  
288 Access Token.

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

### 290 **5.3 Credential overview**

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

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

## 295 **6 Device provisioning for OCF Cloud**

### 296 **6.1 Cloud Provisioning General**

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

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

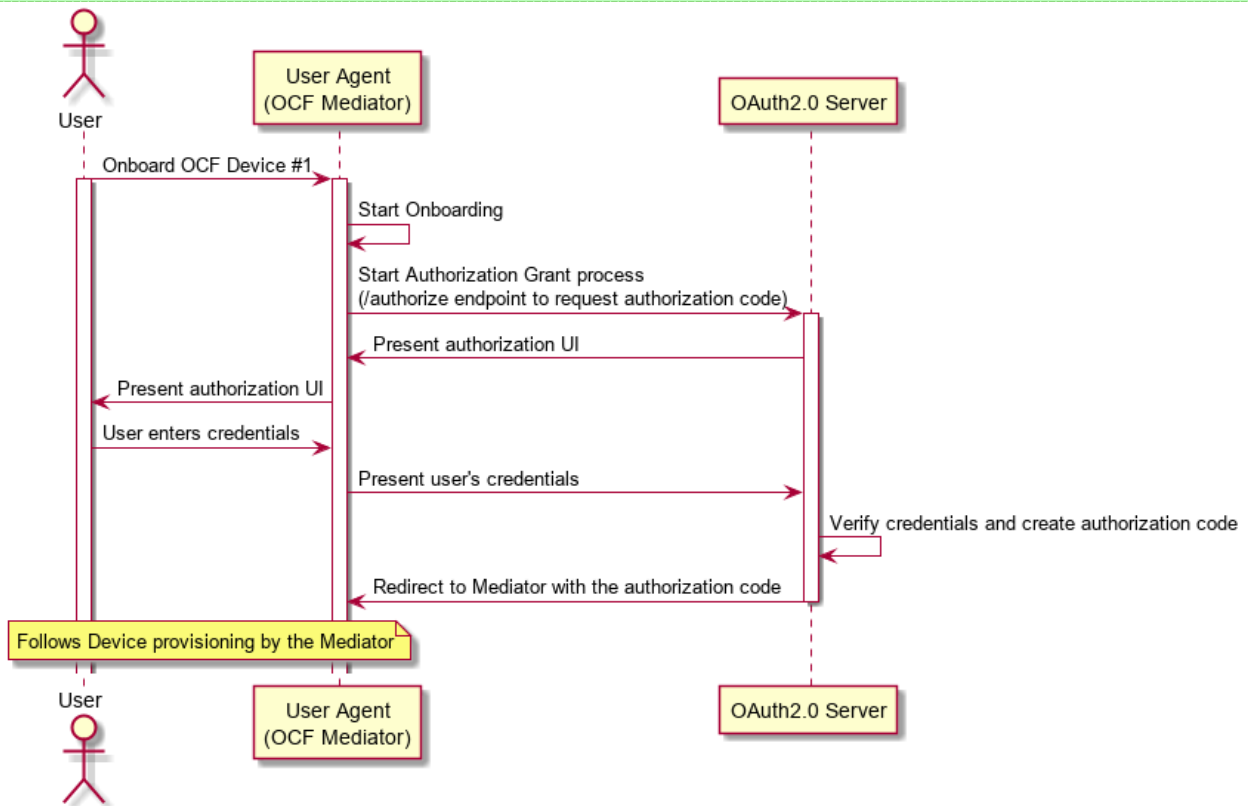
#### 305 **Figure 2 User authorization and provisioning using Authorization Code Grant Flow**

306 @startuml  
307  
308 actor User

```

309 participant UserAgent as "User Agent\n(OCF Mediator)"
310 participant OAuthServer as "OAuth2.0 Server"
311 activate User
312 activate UserAgent
313 UserAgent -> UserAgent: Start Onboarding
314 UserAgent -> OAuthServer: Start Authorization Grant process\n(/authorize endpoint to
315 request authorization code)
316 activate OAuthServer
317 OAuthServer -> UserAgent: Present authorization UI
318 UserAgent -> User: Present authorization UI
319 User -> UserAgent: User enters credentials
320 UserAgent -> OAuthServer: Present user's credentials
321 OAuthServer -> OAuthServer: Verify credentials and create authorization code
322 OAuthServer -> UserAgent: Redirect to Mediator with the authorization code
323 deactivate OAuthServer
324 note over User, UserAgent
325     Follows Device provisioning by the Mediator
326 end note
327
328 @enduml

```



329  
330  
331

## 332 6.2 Device Provisioning by Mediator

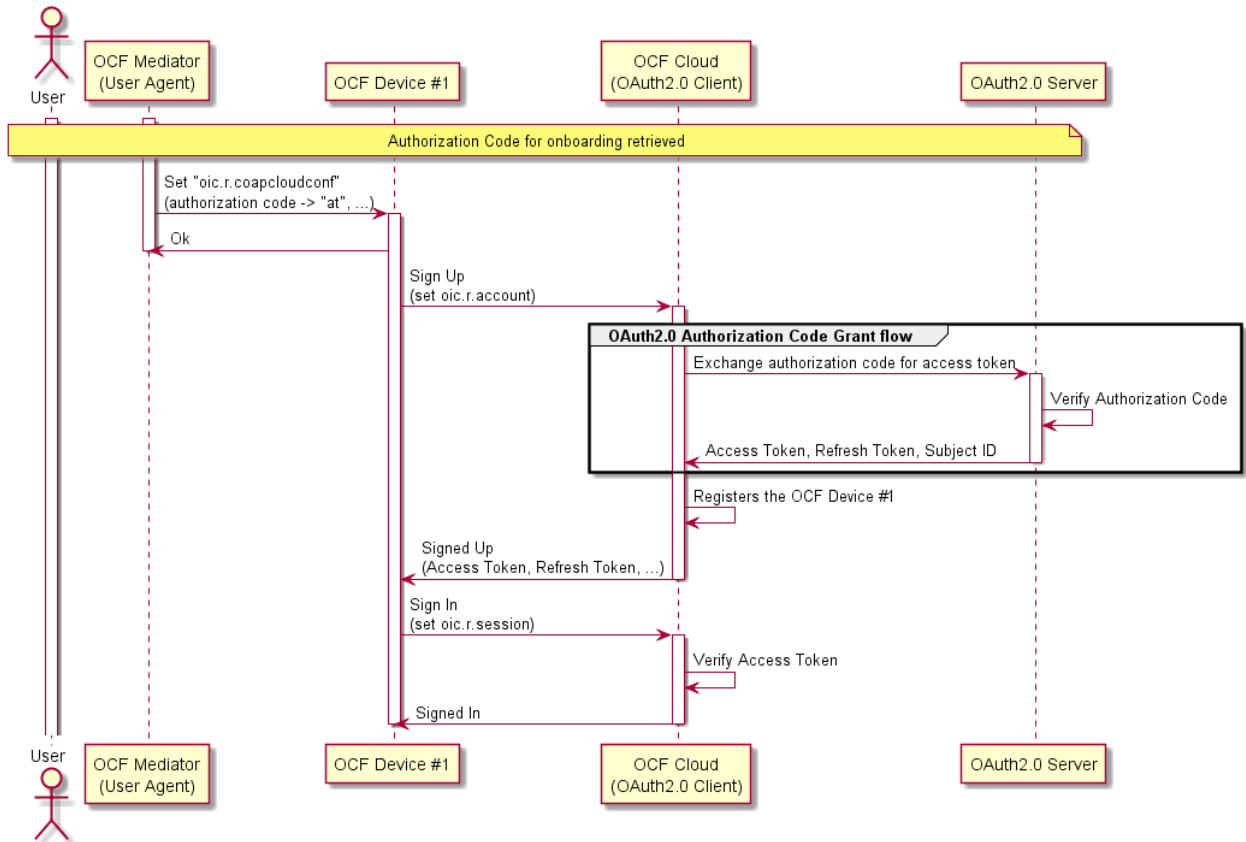
333 The Mediator and the Device shall use the secure session to provision the Device to connect with  
334 the OCF Cloud.

335 The Mediator obtains an Authorization Code or directly an Access Token from the Authorization  
336 Server as described in OCF Cloud Specification. This value is then used by the Device for  
337 registering with the OCF Cloud as described in clause 7. At the time of Device Registration OCF  
338 Cloud exchanges the Authorization Code for the Access Token, returns it back to the OCF Device  
339 and associates the TLS session with corresponding Device ID. The OCF Cloud maintains a map  
340 where Access Token and Mediator provided Device ID are stored.

341 The Mediator provisions the Device, as described in OCF Cloud Specification. The Mediator  
342 provisions OCF Cloud URI to the "cis" Property of "oic.r.coapcloudconf" Resource, OCF Cloud  
343 UUID to the "sid" Property of "oic.r.coapcloudconf" Resource and per-device Access Token or  
344 Authorization Code to the "at" Property of "oic.r.coapcloudconf" Resource on Device. Exchanged  
345 and returned provisioned Access Token is to be treated by Device as an Access Token with  
346 "Bearer" token type as defined in IETF RFC 6750. See Figure 3 for the detailed overview of the  
347 recommended flow, which includes optional OAuth 2.0 Authorization Code Grant

**Figure 3 Device Provisioning using Authorization Code Grant Flow**

```
348 @startuml
349
350
351 actor User
352 participant Mediator as "OCF Mediator\n(User Agent)"
353 participant Device as "OCF Device #1"
354 participant Cloud as "OCF Cloud\n(OAuth2.0 Client)"
355 participant OAuthServer as "OAuth2.0 Server"
356
357 activate User
358 activate Mediator
359
360 note over User, OAuthServer
361     Authorization Code for onboarding retrieved
362 end note
363
364 Mediator -> Device: Set "oic.r.coapcloudconf"\n(authorization code -> "at", ...)
365 activate Device
366 Device -> Mediator: Ok
367 deactivate Mediator
368 Device -> Cloud: Sign Up\n(set oic.r.account)
369 activate Cloud
370 group OAuth2.0 Authorization Code Grant flow
371     Cloud -> OAuthServer: Exchange authorization code for access token
372     activate OAuthServer
373     OAuthServer -> OAuthServer: Verify Authorization Code
374     OAuthServer -> Cloud: Access Token, Refresh Token, Subject ID
375     deactivate OAuthServer
376 end
377 Cloud -> Cloud: Registers the OCF Device #1
378 Cloud -> Device: Signed Up\n(Access Token, Refresh Token, ...)
379 deactivate Cloud
380 Device -> Cloud: Sign In\n(set oic.r.session)
381 activate Cloud
382 Cloud -> Cloud: Verify Access Token
383 Cloud -> Device: Signed In
384 deactivate Device
385 deactivate Cloud
386
387 @enduml
```



388  
389  
390

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

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

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

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

405 The Mediator should provision the "oic.r.coapcloudconf" Resource with the Properties in Table 1.  
406 These details once provisioned are used by the Device to perform Device Registration to the OCF  
407 Cloud. After the initial registration, the Device should use updated values received from the OCF

408 Cloud instead. If OCF Cloud User wants the Device to re-register with the OCF Cloud, they can  
 409 use the Mediator to re-provision the "oic.r.coapcloudconf" Resource with the new values.

410 **Table 1 – Mapping of Properties of the "oic.r.account" and "oic.r.coapcloudconf"**  
 411 **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.

## 412 7 Device authentication with OCF Cloud

### 413 7.1 Device Authentication with OCF Cloud General

414 The mechanisms for Device Authentication in clauses **Error! Reference source not found.**, **Error!**  
 415 **Reference source not found.** and **Error! Reference source not found.** of OCF Security  
 416 Specification imply that a Device is authorized to communicate with any other Device meeting the  
 417 criteria provisioned in "/oic/sec/cred"; the "/oic/sec/acl2" Resource (or "/oic/sec/acl1" resource of  
 418 OIC1.1 Servers) are additionally used to restrict access to specific Resources. The present clause  
 419 describes Device authentication for OCF Cloud, which uses slightly different criteria as described  
 420 in clause 0. A Device accessing an OCF Cloud shall establish a TLS session. The mutual  
 421 authenticated TLS session is established using Server certificate and Client certificate.

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

### 427 7.2 Device Connection with the OCF Cloud

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

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



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

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

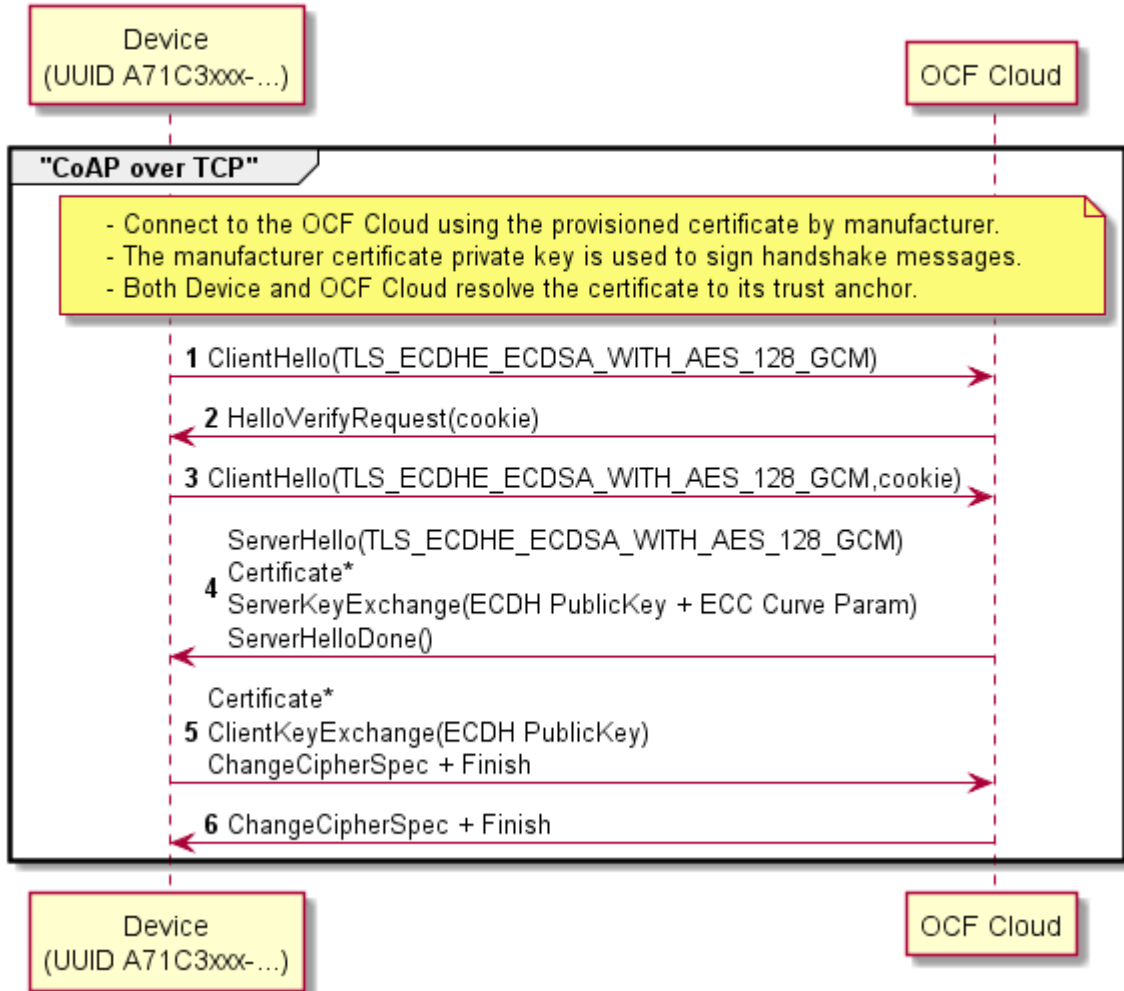
436 The assumption is that the OCF Cloud User trusts the OCF Cloud that the Device connects. The  
437 OCF Cloud connection should not happen without the consent of the OCF Cloud User. The  
438 assumption is that the OCF Cloud User has either service agreement with the OCF Cloud provider  
439 or uses manufacturer provided OCF Cloud.

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

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

```
445 @startuml
446 autonumber
447 title Device Connection with OCF Cloud
448 participant "Device\n(UUID A71C3xxx-...)" as RS
449 participant "OCF Cloud" as CI
450
451 group "CoAP over TCP"
452 note over RS, CI
453 - Connect to the OCF Cloud using the provisioned certificate by manufacturer.
454 - The manufacturer certificate private key is used to sign handshake messages.
455 - Both Device and OCF Cloud resolve the certificate to its trust anchor.
456 end note
457
458 RS->CI: ClientHello(TLS_ECDHE_ECDSA_WITH_AES_128_GCM)
459 CI->RS: HelloVerifyRequest(cookie)
460 RS->CI: ClientHello(TLS_ECDHE_ECDSA_WITH_AES_128_GCM,cookie)
461 CI->RS:
462 ServerHello(TLS_ECDHE_ECDSA_WITH_AES_128_GCM)\nCertificate*\nServerKeyExchange(ECDH
463 PublicKey + ECC Curve Param)\nServerHelloDone()
464 RS->CI: Certificate*\nClientKeyExchange(ECDH PublicKey)\nChangeCipherSpec + Finish
465 CI->RS: ChangeCipherSpec + Finish
466
467 End
468 @enduml
```

**Device Connection with OCF Cloud**



469

470

471

**Figure 4 – Device connection with OCF Cloud**

**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

472

**7.3 Security Considerations**

473

474

475

476

477

478

479

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

480 elevated privileges when submitting a request via the OCF Cloud. In the present document release,  
481 the details of the mechanism are left to the implementation.

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

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

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

## 489 **8 Message integrity and confidentiality**

### 490 **8.1 Cloud Session Semantics**

491 The messages between the OCF Cloud and Device shall be exchanged only if the Device and OCF  
492 Cloud authenticate each other as described in 7. The asymmetric cipher suites as described in 8.2  
493 shall be employed for establishing a secured session and for encrypting/decrypting between the  
494 OCF Cloud and the Device. The OCF Endpoint sending the message shall encrypt and authenticate  
495 the message using the cipher suite as described in 8.2 and the OCF Endpoint shall verify and  
496 decrypt the message before processing it.

### 497 **8.2 Cipher suites for OCF Cloud Credentials**

498 All Devices supporting OCF Cloud Certificate Credentials shall implement:

499 TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256

500 All Devices supporting OCF Cloud Certificate Credentials should implement:

501 TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256,

502 TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA256,

503 TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_GCM\_SHA384,

504 TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CBC\_SHA384,

505 TLS\_ECDHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384

506

## 507 **9 Security resources**

### 508 **9.1 Account Resource**

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

515 During the Device Registration process, an OCF Cloud can provide a distinct URI of another OCF  
516 Cloud ("redirected-to" OCF Cloud). Both initial and redirected-to OCF Clouds are expected to  
517 belong to the same Vendor; they are assumed to have the same UUID and are assumed to have  
518 an out-of-band communication mechanism established. Device does not have to perform the

519 Device Registration on the redirected-to OCF Cloud and the OCF Cloud may ignore such attempts.  
 520 Redirected-to OCF Cloud is expected to accept the Access Token, provided to the Device by the  
 521 initial OCF Cloud.

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

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

- 525 – A Device intending to register with the OCF Cloud shall send UPDATE with following Properties  
 526 "di" ("di" Property Value of "/oic/d" Resource), and "acesstoken" as configured by the Mediator  
 527 ("at" Property Value of "oic.r.coapcloudconf" Resource). The OCF Cloud verifies it is the same  
 528 "acesstoken" which was assigned to the Mediator for the corresponding "di" Property Value.  
 529 The "acesstoken" is the permission for the Device to access the OCF Cloud. If the "apn" was  
 530 included when the Mediator UPDATED the "oic.r.coapcloudconf" Resource, the Device shall  
 531 also include "authprovider" Property when registering with the OCF Cloud. If no "apn" is  
 532 specified, then the "authprovider" Property shall not be included in the UPDATE request.
- 533 – OCF Cloud returns "acesstoken", "uid", "refreshtoken", "expiresin" It may also return  
 534 "redirecturi". Received "acesstoken" is to be treated by Device as an Access Token with  
 535 "Bearer" token type as defined in IETF RFC 6750. This "acesstoken" shall be used for the  
 536 following Account Session start using "oic/sec/session" SVR. Received "refreshtoken" is to be  
 537 treated by Device as a Refresh Token as defined in IETF RFC 6749. The Device stores the  
 538 OCF Cloud's Response values. If "redirecturi" is received, Device shall use received value as  
 539 a new OCF Cloud URI instead of "cis" Property Value of "oic.r.coapcloudconf" Resource for  
 540 further connections.

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

- 542 – To deregister with the OCF Cloud, a DELETE operation shall be sent with the "acesstoken"  
 543 and either "uid", or "di" to be deregistered with the OCF Cloud. On DELETE with the OCF Cloud,  
 544 the Device should also delete values internally stored. Once deregister with an OCF Cloud,  
 545 Device can connect to any other OCF Cloud. Device deregistered need to go through the steps  
 546 in 6 again to be registered with the OCF Cloud.

547 " oic.r.account " Resource is defined in Table 3.

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.base ine	Resource used for a device to add itself under a given credential	N/A

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

550

**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	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 ID, 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 "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 "oic.r.session".

572 **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

### 573 9.3 Account Token Refresh Resource

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

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

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

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

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

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

585 – OCF Cloud response is expected to include a "refreshtoken", new "acesstoken", and  
 586 "expiresin". Received "acesstoken" is to be treated by Device as an Access Token with  
 587 "Bearer" token type as defined in IETF RFC 6750. This Access Token is the permission for the  
 588 Device to access the OCF Cloud. Received "refreshtoken" is to be treated by Device as a  
 589 Refresh Token as defined in IETF RFC 6749. Received "refreshtoken" may be the new Refresh  
 590 Token or the same one as provided by the Device in the UPDATE request. In case when new  
 591 distinct "refreshtoken" is provided by the OCF Cloud, the Device shall discard the old value.  
 592 The OCF Cloud's response values "refreshtoken", "acesstoken" and "expiresin" are securely  
 593 stored on the Device.

594 "oic.r.tokenrefresh" Resource is defined in Table 7.

595 **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

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

597 **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).

## 598 10 Security hardening guidelines

599 In addition to the Sensitive Data list outlined in Table 75 of Security Specification, any Device  
 600 implementing OCF Cloud connection capabilities should also provide reasonable protection for the  
 601 information in Table 9.

602

**Table 9 – Sensitive Data related to OCF Cloud**

<b>Data</b>	<b>Integrity protection</b>	<b>Confidentiality protection</b>
OCF Cloud URL	Yes	Not required
OCF Cloud Identity	Yes	Not required

603

604

605

606

607

DRAFT



608  
 609  
 610  
 611

## Annex A (normative) Resource Type definitions

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

 613 **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

 614 **A.1 Account Token**

 615 **A.1.1 Introduction**

616 Sign-up using generic account provider.

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

618 /oic/sec/account

 619 **A.1.3 Resource type**

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

 621 **A.1.4 OpenAPI 2.0 definition**

```

622 {
623   "swagger": "2.0",
624   "info": {
625     "title": "Account Token",
626     "version": "20190111",
627     "license": {
628       "name": "OCF Data Model License",
629       "url":
630 "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI
631 CENSE.md",
632       "x-copyright": "copyright 2016-2017, 2019 Open Connectivity Foundation, Inc. All rights
633 reserved."
634     },
635     "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
636   },
637   "schemes": ["http"],
638   "consumes": ["application/json"],
639   "produces": ["application/json"],
640   "paths": {
641     "/oic/sec/account" : {
642       "post": {
643         "description": "Sign-up using generic account provider.\n",
644         "parameters": [
645           { "$ref": "#/parameters/interface" },
646           {
647             "name": "body",
648             "in": "body",
649             "required": true,
650             "schema": { "$ref": "#/definitions/Account-request" },
651             "x-example":
652 {

```

```

653         "di" : "9cfbeb8e-5ale-4d1c-9d01-00c04fd430c8",
654         "authprovider" : "github",
655         "accesstoken" : "8802f2eaf8b5e147a936"
656     }
657 }
658 ],
659 "responses": {
660     "204": {
661         "description" : "2.04 Changed respond with required and optional information\n",
662         "x-example":
663         {
664             "rt": ["oic.r.account"],
665             "accesstoken" : "0f3d9f7fe5491d54077d",
666             "refreshToken" : "00fe4644a6fbe5324eec",
667             "expiresin" : 3600,
668             "uid" : "123e4567-e89b-12d3-a456-d6e313b71d9f",
669             "redirecturi" : "coaps+tcp://example.com:443"
670         },
671         "schema": { "$ref": "#/definitions/Account-response" }
672     }
673 }
674 },
675 "delete": {
676     "description": "Delete a device. This also removes all resources in the device on cloud
677 side.\nexample: /oic/account?di=9cfbeb8e-5ale-4d1c-9d01-
678 00c04fd430c8&accesstoken=0f3d9f7fe5491d54077d\n",
679     "parameters": [
680         {"$ref": "#/parameters/interface"}
681     ],
682     "responses": {
683         "202": {
684             "description" : "2.02 Deleted response informing the device is successfully
685 deleted.\n"
686         }
687     }
688 }
689 }
690 },
691 "parameters": {
692     "interface": {
693         "in" : "query",
694         "name" : "if",
695         "type" : "string",
696         "enum" : ["oic.if.baseline"]
697     }
698 },
699 "definitions": {
700     "Account-request" : {
701         "properties": {
702             "authprovider": {
703                 "description": "The name of Authorization Provider through which Access Token was
704 obtained",
705                 "type": "string"
706             },
707             "accesstoken" : {
708                 "description": "Access-Token used for communication with OCF Cloud after account
709 creation",
710                 "pattern": "(?!$|\\s+).*",
711                 "type": "string"
712             },
713             "di": {
714                 "description": "Format pattern according to IETF RFC 4122.",
715                 "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
716 9]{12}$",
717                 "type": "string"
718             }
719         }
720     },

```

```

720     "type" : "object",
721     "required": ["di", "accesstoken"]
722 },
723 "Account-response": {
724   "properties": {
725     "expiresin" : {
726       "description": "Access-Token remaining life time in seconds (-1 if permanent)",
727       "readOnly": true,
728       "type": "integer"
729     },
730     "rt": {
731       "description": "Resource Type of the Resource",
732       "items": {
733         "maxLength": 64,
734         "type": "string",
735         "enum" : ["oic.r.account"]
736       },
737       "minItems": 1,
738       "maxItems": 1,
739       "readOnly": true,
740       "type": "array"
741     },
742     "refreshtoken" : {
743       "description": "Refresh token can be used to refresh the Access Token before getting
744 expired",
745       "pattern": "(?!$|\\s+).*",
746       "readOnly": true,
747       "type": "string"
748     },
749     "uid" : {
750       "description": "Format pattern according to IETF RFC 4122.",
751       "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
752 9]{12}$",
753       "type": "string"
754     },
755     "accesstoken" : {
756       "description": "Access-Token used for communication with cloud after account creation",
757       "pattern": "(?!$|\\s+).*",
758       "type": "string"
759     },
760     "n": {
761       "$ref":
762 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
763 schema.json#/definitions/n"
764     },
765     "id": {
766       "$ref":
767 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
768 schema.json#/definitions/id"
769     },
770     "redirecturi" : {
771       "description": "Using this URI, the Client needs to reconnect to a redirected OCF Cloud.
772 If provided, this value shall be used by the Device instead of Mediator-provided URI during the
773 Device Registration.",
774       "readOnly": true,
775       "type": "string"
776     },
777     "if": {
778       "description": "The interface set supported by this resource",
779       "items": {
780         "enum": [
781           "oic.if.baseline"
782         ],
783         "type": "string"
784       },
785       "minItems": 1,
786       "maxItems": 1,

```

```

787         "uniqueItems": true,
788         "readOnly": true,
789         "type": "array"
790     },
791 },
792 "type" : "object",
793 "required": ["accesstoken", "refreshtoken", "expiresin", "uid"]
794 }
795 }
796 }
797

```

### 798 A.1.5 Property definition

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

800 **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.
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, 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: schema	see	No	Read Only	Resource Type of the Resource
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.
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: schema	see	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	

### 801 A.1.6 CRUDN behaviour

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

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

Create	Read	Update	Delete	Notify
		post	delete	

## 804 A.2 Session

### 805 A.2.1 Introduction

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

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

808 /oic/sec/session

809 **A.2.3 Resource type**

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

811 **A.2.4 OpenAPI 2.0 definition**

```
812 {
813   "swagger": "2.0",
814   "info": {
815     "title": "Session",
816     "version": "v1.0-20181001",
817     "license": {
818       "name": "OCF Data Model License",
819       "url":
820 "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI
821 CENSE.md",
822     "x-copyright": "copyright 2016-2017, 2019 Open Connectivity Foundation, Inc. All rights
823 reserved.",
824   },
825   "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
826 },
827   "schemes": ["http"],
828   "consumes": ["application/json"],
829   "produces": ["application/json"],
830   "paths": {
831     "/oic/sec/session" : {
832       "post": {
833         "description": "Resource that manages the persistent session between a Device and OCF
834 Cloud.",
835         "parameters": [
836           {"$ref": "#/parameters/interface"},
837           {
838             "name": "body",
839             "in": "body",
840             "required": true,
841             "schema": { "$ref": "#/definitions/Account-Session-Request" },
842             "x-example":
843             {
844               "uid" : "123e4567-e89b-12d3-a456-d6e313b71d9f",
845               "di" : "9cfbeb8e-5a1e-4d1c-9d01-00c04fd430c8",
846               "accesstoken" : "0f3d9f7fe5491d54077d",
847               "login" : true
848             }
849           }
850         ],
851         "responses": {
852           "204": {
853             "description": "",
854             "x-example":
855             {
856               "rt": ["oic.r.session"],
857               "expiresin" : 3600
858             },
859           "schema": { "$ref": "#/definitions/Account-Session-Response" }
860         }
861       }
862     }
863   },
864 },
865   "parameters": {
866     "interface" : {
867       "in" : "query",
868       "name" : "if",
869       "type" : "string",
```

```

870     "enum" : ["oic.if.baseline"]
871   }
872 },
873 "definitions": {
874   "Account-Session-Request" : {
875     "properties": {
876       "uid": {
877         "description": "Format pattern according to IETF RFC 4122.",
878         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
879         "type": "string"
880       },
881       "di": {
882         "description": "The Device ID\nFormat pattern according to IETF RFC 4122.",
883         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
884         "type": "string"
885       },
886       "accesstoken": {
887         "description": "Access-Token used to grant access right for the Device to sign-in.",
888         "pattern": "(?!$|\\s+).*",
889         "type": "string"
890       },
891       "login": {
892         "description": "Action for the request: true = login, false = logout.",
893         "type": "boolean"
894       }
895     }
896   },
897   "type" : "object",
898   "required": ["uid", "di", "accesstoken", "login"]
899 },
900 "Account-Session-Response" : {
901   "properties": {
902     "expiresin": {
903       "description": "Access-Token remaining life time in seconds (-1 if permanent).",
904       "readOnly": true,
905       "type": "integer"
906     },
907     "rt": {
908       "description": "Resource Type of the Resource.",
909       "items": {
910         "maxLength": 64,
911         "type": "string",
912         "enum": ["oic.r.session"]
913       },
914       "minItems": 1,
915       "readOnly": true,
916       "type": "array"
917     },
918     "n": {
919       "$ref":
920       "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-schema.json#/definitions/n"
921     },
922     "id": {
923       "$ref":
924       "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-schema.json#/definitions/id"
925     },
926     "if": {
927       "description": "The interface set supported by this Resource.",
928       "items": {
929         "enum": [
930           "oic.if.baseline"
931         ],
932         "type": "string"
933       }
934     }
935   }
936 }

```

```

937     "minItems": 1,
938     "readOnly": true,
939     "type": "array"
940   },
941 },
942 "type" : "object",
943 "required" : ["expiresin"]
944 }
945 }
946 }
947

```

### 948 A.2.5 Property definition

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

950 **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.

## 951 A.2.6 CRUDN behaviour

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

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

Create	Read	Update	Delete	Notify
		post		

## 954 A.3 Token Refresh

### 955 A.3.1 Introduction

956 Obtain fresh Access Token using the refresh token, client should refresh Access Token before it  
 957 expires.

### 958 A.3.2 Well-known URI

959 /oic/sec/tokenrefresh

### 960 A.3.3 Resource type

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

### 962 A.3.4 OpenAPI 2.0 definition

```

963 {
964   "swagger": "2.0",
965   "info": {
966     "title": "Token Refresh",
967     "version": "v1.0-20181001",
968     "license": {
969       "name": "OCF Data Model License",
970       "url":
971 "https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LI
972 CENSE.md",
973       "x-copyright": "copyright 2016-2017, 2019 Open Connectivity Foundation, Inc. All rights
974 reserved."
975     },
976     "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
977   },
978   "schemes": ["http"],
979   "consumes": ["application/json"],
980   "produces": ["application/json"],
981   "paths": {
982     "/oic/sec/tokenrefresh" : {
983       "post": {
984         "description": "Obtain fresh access-token using the refresh token, client should refresh
985 access-token before it expires.\n",

```

```

986     "parameters": [
987       { "$ref": "#/parameters/interface" },
988       {
989         "name": "body",
990         "in": "body",
991         "required": true,
992         "schema": { "$ref": "#/definitions/TokenRefresh-Request" },
993         "x-example":
994           {
995             "uid" : "123e4567-e89b-12d3-a456-d6e313b71d9f",
996             "di" : "9cfbeb8e-5ale-4dlc-9d01-00c04fd430c8",
997             "refreshtoken" : "00fe4644a6fbe5324eec"
998           }
999       }
1000     ],
1001     "responses": {
1002       "204": {
1003         "description": "2.04 Changed respond with new access-token.\n",
1004         "x-example":
1005           {
1006             "rt": ["oic.r.tokenrefresh"],
1007             "accesstoken" : "8ce598980761869837be",
1008             "refreshtoken" : "d4922312b6df0518e146",
1009             "expiresin" : 3600
1010           }
1011         ,
1012         "schema": { "$ref": "#/definitions/TokenRefresh-Response" }
1013       }
1014     }
1015   }
1016 }
1017 },
1018 "parameters": {
1019   "interface" : {
1020     "in" : "query",
1021     "name" : "if",
1022     "type" : "string",
1023     "enum" : ["oic.if.baseline"]
1024   }
1025 },
1026 "definitions": {
1027   "TokenRefresh-Request" : {
1028     "properties": {
1029       "refreshtoken": {
1030         "description": "Refresh token received by account management or during token refresh
1031 procedure.",
1032         "pattern": "(?!$|\\s+).*",
1033         "type": "string"
1034       },
1035       "uid": {
1036         "description": "Format pattern according to IETF RFC 4122.",
1037         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
1038 9]{12}$",
1039         "type": "string"
1040       },
1041       "di": {
1042         "description": "Format pattern according to IETF RFC 4122.",
1043         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
1044 9]{12}$",
1045         "type": "string"
1046       }
1047     },
1048     "type": "object",
1049     "required": ["uid", "di", "refreshtoken"]
1050   },
1051   "TokenRefresh-Response" : {
1052     "properties": {

```

```

1053     "expiresin": {
1054         "description": "Access-Token life time in seconds (-1 if permanent).",
1055         "readOnly": true,
1056         "type": "integer"
1057     },
1058     "rt": {
1059         "description": "Resource Type of the Resource.",
1060         "items": {
1061             "maxLength": 64,
1062             "type": "string",
1063             "enum": ["oic.r.tokenrefresh"]
1064         },
1065         "minItems": 1,
1066         "readOnly": true,
1067         "type": "array"
1068     },
1069     "refreshtoken": {
1070         "description": "Refresh token received by account management or during token refresh
1071 procedure.",
1072         "pattern": "(?!$|\\s+).*",
1073         "type": "string"
1074     },
1075     "accesstoken": {
1076         "description": "Granted Access-Token.",
1077         "pattern": "(?!$|\\s+).*",
1078         "readOnly": true,
1079         "type": "string"
1080     },
1081     "n": {
1082         "$ref":
1083 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1084 schema.json#/definitions/n"
1085     },
1086     "id": {
1087         "$ref":
1088 "https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
1089 schema.json#/definitions/id"
1090     },
1091     "if" :
1092     {
1093         "description": "The interface set supported by this Resource.",
1094         "items": {
1095             "enum": [
1096                 "oic.if.baseline"
1097             ],
1098             "type": "string"
1099         },
1100         "minItems": 1,
1101         "readOnly": true,
1102         "type": "array"
1103     }
1104 },
1105 "type" : "object",
1106 "required": ["accesstoken", "refreshtoken", "expiresin"]
1107 }
1108 }
1109 }
1110

```

### 1111 A.3.5 Property definition

1112 Table A.6 defines the Properties that are part of the "oic.r.tokenrefresh" Resource Type.

1113 **Table A.6 – The Property definitions of the Resource with type "rt" = "oic.r.tokenrefresh".**

Property name	Value type	Mandatory	Access mode	Description
---------------	------------	-----------	-------------	-------------

refresh token	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.
refresh token	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	

 1114 **A.3.6 CRUDN behaviour**

 1115 Table A.7 defines the CRUDN operations that are supported on the "oic.r.tokenrefresh" Resource  
 1116 Type.

1117 **Table A.7 – The CRUDN operations of the Resource with type "rt" = "oic.r.tokenrefresh".**1118  
1119  

<b>Create</b>	<b>Read</b>	<b>Update</b>	<b>Delete</b>	<b>Notify</b>
		post		

DRAFT