



1 **FIDO Technical Glossary**

2 **Specification Set: fido-v1.0-rd-20140209 REVIEW DRAFT**

3 **Editors:**

4 Rolf Lindeman, Nok Nok Labs

5 Davit Baghdasaryan, Nok Nok Labs

6 Brad Hill, PayPal

7 **Contributors:**

8 Jeff Hodges, PayPal

9 **Abstract:**

10 This document defines many of the technical terms and phrases used in
11 FIDO Alliance specifications and documents.

12 **Status:**

13 This Specification has been prepared by FIDO Alliance, Inc. **This is a Review Draft Specification and**
14 **is not intended to be a basis for any implementations as the Specification may change.** Permission is
15 hereby granted to use the Specification solely for the purpose of reviewing the Specification. No rights
16 are granted to prepare derivative works of this Specification. Entities seeking permission to reproduce
17 portions of this Specification for other uses must contact the FIDO Alliance to determine whether an ap-
18 propriate license for such use is available.

19 Implementation of certain elements of this Specification may require licenses under third party intellec-
20 tual property rights, including without limitation, patent rights. The FIDO Alliance, Inc. and its Members
21 and any other contributors to the Specification are not, and shall not be held, responsible in any manner
22 for identifying or failing to identify any or all such third party intellectual property rights.

23 THIS FIDO ALLIANCE SPECIFICATION IS PROVIDED “AS IS” AND WITHOUT ANY WAR-
24 RANTY OF ANY KIND, INCLUDING, WITHOUT LIMITATION, ANY EXPRESS OR IMPLIED
25 WARRANTY OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICU-
26 LAR PURPOSE.

27 Copyright © 2014 FIDO Alliance, Inc. All rights reserved.

Table of Contents

[1 Notation.....4](#)
 [1.1 Key Words.....4](#)
 [1.2 Revision History.....4](#)
[2 Introduction.....5](#)
[3 Definitions.....6](#)
[Bibliography.....18](#)



28 1 Notation

29 Type names, attribute names and element names are written in *italics*.

30 String literals are enclosed in “”, e.g. “UAF-TLV”.

31 In formulas we use “|” to denote byte wise concatenation operations.

32 1.1 Key Words

33 The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”,
34 “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this doc-
35 ument are to be interpreted as described in [RFC2119].

36 1.2 Revision History

37 This revision history may be subsumed by the SVN checkin comments and/or JIRA
38 comments once that is integrated.

39 In any case, I would expect this section to disappear as part of the publication process.

40 2 Introduction

41 This document is the FIDO Alliance glossary of normative technical terms.

42 This document is not an exhaustive compendium of all FIDO technical terminology because the FIDO terminology is built upon existing terminology.

43
44 Thus many terms that are commonly used within this context are not listed.

45 They may be found in the glossaries/documents/specifications referenced in
46 the bibliography. Terms defined here that are not attributed to other glossaries/documents/specifications are being defined here.

47
48 This glossary is expected to evolve along with the FIDO Alliance specifications and documents.
49

50 3 Definitions

51 **AAID**

52 Authenticator Attestation ID. See *Attestation ID*.

53 **Application**

54 A set of functionality provided by a common entity (the application owner, aka the
55 *Relying Party*), and perceived by the user as belonging together.

56 **Application Facet**

57 An (application) facet is how an application is implemented on various platforms.
58 For example, the application MyBank may have an Android app, an iOS app, and
59 a Web app. These are all facets of the MyBank application.

60 **Application Facet ID**

61 A platform-specific identifier (URI) for an application facet.

- 62 ○ For Web applications, the facet id is the RFC 6454 origin.
- 63 ○ For Android applications, the facet id is the URI android:apk-key-
64 hash:<hash-of-apk-signing-cert>
- 65 ○ For iOS, the facet id is the URI ios:bundle-id:<ios-bundle-id-of-app>

66 **AppID**

67 The AppID is an identifier for a set of different Facets of a relying party's applica-
68 tion. The AppID is a URL pointing to the TrustedApps, i.e. list of FacetIDs related
69 to this AppID.

70 **Attestation**

71 In the FIDO context, attestation is how Authenticators make claims to a Relying
72 Party that the keys they generate, and/or certain measurements they report, orig-
73 inate from genuine devices with certified characteristics.

74 **Attestation Certificate**

75 A public key certificate related to an Attestation Key.

76 **Authenticator Attestation ID / AAID**

77 A unique identifier assigned to a model, class or batch of FIDO Authenticators
78 that all share the same characteristics, and which a Relying Party can use to look
79 up an Attestation Public Key and Authenticator Metadata for the device.

80 **Attestation [Public / Private] Key**

81 A key used for FIDO Authenticator attestation.

82 **Attestation Root Certificate**

83 A root certificate explicitly trusted by the FIDO Alliance, to which Attestation Cer-
84 tificates chain to.

85 **Authentication**

86 Authentication is the process in which user employs their FIDO Authenticator to
87 prove possession of a registered key to a relying party.

88 **Authentication Algorithm**

89 The combination of signature and hash algorithms used for authenticator-to-rely-
90 ing party authentication.

91 **Authentication Scheme**

92 The combination of an Authentication Algorithm with a message syntax or fram-
93 ing that is used by an Authenticator when constructing a response.

94 **Authenticator, Authnr**

95 See *FIDO Authenticator*.

96 **Authenticator, 1stF / First Factor**

97 A FIDO Authenticator that transactionally provides a username and at least two
98 authentication factors: cryptographic key material (something you have) plus user
99 verification (something you know / something you are) and so can be used by it-
100 self to complete an authentication.

101 It is assumed that these authenticators have an internal matcher. The matcher is
102 able to verify an already enrolled user. If there is more than one user enrolled –
103 the matcher is also able to identify the right user.

104 Examples of such authenticator is a biometric sensor or a PIN based verification.
105 Authenticators which only verify presence (e.g. a physical button) or perform no
106 verification at all cannot act as 1stF Authenticator.

107 **Authenticator, 2ndF / Second Factor**

108 A FIDO Authenticator which acts only as a second factor. 2ndF Authenticators
109 always require a single Key Handle to be provided before responding to a Sign
110 command. They might or might not have a user verification method.

111 It is assumed that these authenticators MAY or MAY not have an internal
112 matcher.

113 **Authenticator Attestation**

114 The process of communicating a cryptographic assertion to a Relying Party that
115 a key presented during Registration was created and protected by a genuine Au-
116 thenticator with verified characteristics.

117 **Authenticator Metadata**

118 Verified information about the characteristics of a certified Authenticator, associ-
119 ated with an AAID and available from the FIDO Alliance. FIDO Servers are ex-
120 pected to have access to up-to-date metadata to be able to interact with a given
121 Authenticator.

122 **Authenticator Policy**

123 A JSON data structure that allows a Relying Party to communicate to a FIDO
124 Client the capabilities or specific authenticators that are allowed or disallowed for
125 use in a given operation.

126 **ASM / Authenticator Specific Module**

127 Software associated with a FIDO Authenticator that provides a uniform interface
128 between the hardware and FIDO Client software.

129 **AV**

130 ASM Version

131 **Bound Authenticator**

132 A FIDO Authenticator or Authenticator + ASM combination which uses an access
133 control mechanism to restrict the use of registered keys to trusted FIDO Clients
134 and/or trusted FIDO User Devices. Compare to *Roaming Authenticator*.

135 **Certificate**

136 An X.509v3 certificate defined by the profile specified in RFC5280 and its suc-
137 cessors. [<http://www.ietf.org/rfc/rfc5280.txt>]

138 **Channel Binding**

139 See: <http://tools.ietf.org/html/rfc5056> and [http://tools.ietf.org/html/draft-balfanz-tls-
140 channelid-01](http://tools.ietf.org/html/draft-balfanz-tls-channelid-01)

141 A channel binding allows applications to establish that the two end-points of a se-
142 cure channel at one network layer are the same as at a higher layer by binding
143 authentication to the higher layer to the channel at the lower layer.

144 **Client**

145 This term is used “in context”, and may refer to a FIDO client or some other type
146 of client, e.g. a TLS client. See *FIDO Client*.

147 **Correlation Handle**

148 Any piece of information that may allow, in the context of FIDO protocols, implicit
149 or explicit association and or attribution of multiple actions, believed by the user
150 to be distinct and unrelated, back to a single unique entity. An example of a cor-
151 relation handle outside of the FIDO context is a client certificate used in tradi-
152 tional TLS mutual authentication: because it sends the same data to multiple Re-
153 lying Parties, they can therefore collude to uniquely identify and track the user
154 across unrelated activities.

155 **Deregistration**

156 A phase of a FIDO protocol in which a Relying Party tells a FIDO Authenticator to
157 forget a specified piece of (or all) locally managed key material associated with a
158 specific Relying Party account, in case such keys are no longer considered valid
159 by the Relying Party.

160 **Discovery**

161 A phase of a FIDO protocol in which a Relying Party is able to determine the
162 availability of FIDO capabilities at the client's device, including metadata about
163 the available authenticators.

164 **E(K,D)**

165 Denotes the Encryption of data D with key K

166 **Enrollment**

167 The process of making a User known to an Authenticator. This might be a Bio-
168 metric Enrollment as defined in (<http://biometrics.gov/Documents/Glossary.pdf>)
169 or involve processes such as taking ownership of and setting a PIN or password
170 for a non-biometric cryptographic storage device. Enrollment may happen as
171 part of a FIDO protocol ceremony, or it may happen outside of the FIDO context
172 for multi-purpose authenticators.

173 **Facet**

174 See *Application Facet*

175 **Facet ID**

176 See *Application Facet ID*

177 **FIDO Authenticator**

178 An Authentication entity that meets the FIDO Alliance's requirements and which
179 has published metadata.

180 A FIDO Authenticator is responsible for *User Verification* and maintaining the
181 cryptographic material required for the Relying Party *Authentication*.

182 It is important to note that a FIDO Authenticator is only considered such for and
183 in relation to its participation in FIDO Alliance protocols. Because the FIDO Al-
184 liance aims to utilize a diversity of existing and future hardware, many devices
185 used for FIDO may have other primary or secondary uses. To the extent that a
186 device is used for non-FIDO purposes such as local operating system login or
187 network login with non-FIDO protocols, it is not considered a FIDO Authenticator
188 and its operation in such modes is NOT subject to FIDO Alliance guidelines or re-
189 strictions, including those related to security and privacy.

190 A FIDO Authenticator may be referred to as simply an Authenticator or abbrevi-
191 ated as "Authnr". Important distinctions in an Authenticator's capabilities and

192 user experience may be experienced depending on whether it is a *Roaming* or
193 *Bound* authenticator, and whether it is a “First Factor” or “Second Factor” authen-
194 ticator.

195 **FIDO Client**

196 This is the software entity processing the UAF or U2F protocol messages on the
197 FIDO User Device. FIDO Clients may take one of two forms:

- 198 ○ A software component implemented in a User Agent (either web browser
199 or native application).
- 200 ○ A standalone piece of software shared by several User Agents. (Web
201 browsers or native applications).

202 **FIDO Data / FIDO Information**

203 Any information gathered or created as part of completing a FIDO transaction.

204 This includes but is not limited to, biometric measurements of or templates for the
205 user and FIDO transaction history.

206 **FIDO Plugin**

207 The implementation of the interface in a web browser that brokers messages be-
208 tween a client side web application and FIDO client. This component is referred
209 to as a “plugin” even if the APIs are built natively into the web browser or injected
210 into a hosted browser component.

211 **FIDO Server**

212 Server software typically deployed in Relying Party’s infrastructure that meets the
213 UAF protocol’s server requirements.

214 **FIDO UAF Client**

215 See *FIDO Client*.

216 **FIDO User Device**

217 The computing device where the FIDO Client operates and from which the user
218 initiates an action that utilizes FIDO.

219 **KeyID**

220 KeyID identifies a registered key between an Authenticator and a FIDO Server
221 for 1F Authenticators. It is used in concert with AAID to identify a particular Au-

222 authenticator that holds the necessary key. KeyID is the SHA256 hash of the Key-
223 Handle managed by the ASM.

224 **KeyHandle**

225 A key container created by a FIDO Authenticator, containing a private key and
226 (optionally) other data (such as Username). A key handle may be wrapped (en-
227 crypted with a key known only to the authenticator) or unwrapped. In the un-
228 wrapped form it is referred to as a Raw Key Handle. 2F Authenticators must re-
229 trieve their Key Handles from the Relying Party to function, 1F Authenticators
230 manage the storage of their own Key Handles, either internally (for External Au-
231 thenticators) or at the ASM layer. (for Internal Authenticators)

232 **Key Registration**

233 The process of securely establishing a key between FIDO Server and FIDO Au-
234 thenticator.

235 **KeyRegistrationData (KRD)**

236 A KeyRegistrationData object is created and returned by an Authenticator as the
237 result of the Authenticator's Register command. The KRD object contains items
238 such as the authenticator's AAID, the newly generated UAuth.pub key, as well as
239 other authenticator-specific information such as algorithms used by the authenti-
240 cator for performing cryptographic operations, and counter values. The KRD ob-
241 ject is signed using the Authenticator's attestation private key.

242 **KHAccessToken**

243 A secret value that acts as a guard for Authenticator Commands. KHAccessTo-
244 kens are generated and provided by an ASM.

245 **Matcher**

246 A component of a FIDO Authenticator which is able to perform local User Verifi-
247 cation. (biometric matching, PIN verification, etc.)

248 **Persona**

249 With the concept of Persona, all relevant data in an Authenticator (e.g. keys) are
250 related to one Persona (e.g. "business" or "personal"). Some administrative inter-
251 face (not standardized by FIDO) of the Authenticator allows maintaining and
252 switching Personas.

253 The User can switch to the “Personal” Persona and register new accounts. After
254 switching back to “Business” Persona, these accounts will not be recognized by
255 the Authenticator (until the User switches back to “Personal” Persona again).

256 **PersonalID**

257 An identifier provided by an ASM, PersonalID is used to associate different regis-
258 trations. It can be used to create virtual identities on a single authenticator, for
259 example to differentiate “personal” and “business” accounts. PersonalIDs can be
260 used to manage privacy settings on the Authenticator.

261 **Roaming Authenticator**

262 A FIDO Authenticator configured to move between different FIDO Clients and
263 FIDO User Devices lacking an established trust relationship by:

- 264 1) Using only its own internal storage for registrations
- 265 2) Allowing registered keys to be employed without access control mecha-
266 nisms at the API layer. (Roaming Authenticators still may perform *User*
267 *Verification.*)

268 Compare to *Bound Authenticator*.

269 **Registration**

270 A phase of a FIDO protocol in which a user generates and associates new key
271 material with an account at the Relying Party, subject to policy set by the server
272 and acceptable attestation that the authenticator and registration matches that
273 policy.

274 **Registration Scheme**

275 The Registration Scheme defines how the authentication key is being exchanged
276 between the FIDO Server and the FIDO Authenticator.

277 **Relying Party**

278 A web site or other entity that uses a FIDO protocol to directly authenticate users
279 (i.e., performs peer-entity authentication). Note that if FIDO is composed with
280 Federated Identity Management protocols (e.g., SAML, OpenID Connect, etc.),
281 the Identity Provider will also be playing the role of a FIDO Relying Party.

282 **S(K, D)**

283 Signing of data D with key K

284 **Secure Display**

285 This is a feature of FIDO Authenticators able to show content of a message to a
286 user and protect the integrity of this message.

287 **Server Challenge**

288 A random value provided by the FIDO Server in the UAF protocol requests.

289 **Sign Counter**

290 A monotonically increasing counter maintained by the Authenticator. It is in-
291 creased on every use of the Uauth (private) key. This value can be used by the
292 FIDO Server to detect cloned Authenticators.

293 **SignedData**

294 A SignedData object is created and returned by an Authenticator as the result of
295 the Authenticator's Sign command. The to-be-signed data input to the Sign com-
296 mand is represented in the returned SignedData object as intact values or as
297 hashed values. The SignedData object also contains general information about
298 the authenticator and its mode, a nonce, information about authenticator-specific
299 cryptographic algorithms, and a use counter. The SignedData object is signed
300 using the Relying Party-specific UAuth.priv key.

301 **Silent Authenticator**

302 FIDO Authenticator that does not prompt the user or perform any *User Verifica-*
303 *tion*.

304 **Template**

305 A biometric template (also called template) is a digital reference of distinct char-
306 acteristics that have been extracted from a biometric sample. Templates are
307 used during the biometric authentication process.

308 **TLS**

309 Transport Layer Security

310 **Token**

311 In U2F, the term Token is often used to mean what is called an Authenticator in
312 UAF. Also, note that other uses of “token”, e.g. KHAccessToken, User Verifica-
313 tion Token, etc., are separately distinct. If they are not explicitly defined, their
314 meaning needs to be determined from context.

315 **Transaction Confirmation**

316 An operation in the FIDO protocol that allows a Relying Party to request that a
317 FIDO Client and Authenticator with the appropriate capabilities display some in-
318 formation to the user, request that the user authenticate locally to their FIDO Au-
319 thenticator to confirm it, and provide proof of possession of previously registered
320 key material an attestation of the confirmation back to the Relying Party.

321 **TrustedApps**

322 The data structure holding the list of FacetIDs. The AppID is used to retrieve this
323 data structure.

324 **TTEXT**

325 Transaction Text, i.e. text to be confirmed in the case of Transaction Confirma-
326 tion.

327 **U2F**

328 Universal 2nd Factor. The FIDO protocol and family of Authenticators to enable
329 a cloud service to offer its users the options of using an easy-to-use, strongly-
330 secure open standards-based 2nd factor device for authentication. It relies on
331 the server to know the (expected) user before triggering the authentication.

332 **UAF**

333 Universal Authentication Framework. The FIDO Protocol and family of Authenti-
334 cators to enable a service to offer its users flexible and interoperable authentica-
335 tion. It allows triggering the authentication before the server knows the user.

336 **UAF Client**

337 See *FIDO Client*.

338 **UAuth.pub / UAuth.priv / UAuth.key**

339 User authentication keys generated by FIDO Authenticator. UAuth.pub is the
340 public part of key pair. UAuth,priv is the private part of the key. UAuth.key is the
341 more generic notation to refer to UAuth.priv.

342 **UINT16**

343 A 16 bit (2 bytes) unsigned integer.

344 **UINT32**

345 A 32 bit (4 bytes) unsigned integer.

346 **UINT64**

347 A 64 bit (8 bytes) unsigned integer.

348 **UPV**

349 UAF Protocol Version

350 **User**

351 Relying Party's user, and owner of the FIDO Authenticator.

352 **User Agent**

353 The user agent is a client application that is acting on behalf of a user in a client-
354 server system. Examples of user agents include web browsers and mobile apps.

355 **User Verification**

356 The process by which a FIDO Authenticator locally authorizes use of key mate-
357 rial, e.g. through a touch, pin code, fingerprint match or other biometric.

358 **User Verification Token**

359 User Verification Token is a token generated by Authenticator and handed to
360 ASM after successful user verification. Without having this token ASM cannot in-
361 voke special commands such as Register or Sign.

362 The lifecycle of User Verification Token is managed by Authenticator. The con-
363 crete technique for generating such token and managing its lifecycle is vendor
364 specific and non-normative.

365 **Username**

366 A human-readable string identifying a user's account at a Relying Party.

367 **Verification Factor**

368 The specific means by which local user verification is accomplished. e.g. finger-
369 print, voiceprint, or PIN.

370 **Web Application, Client-Side**

371 The portion of a Relying Party application built on the Open Web Platform which
372 executes in the User Agent. When the term "Web Application" appears unquali-
373 fied or without specific context in FIDO documents, it generally refers to either
374 the client-side portion or the combination of both client-side and server-side
375 pieces of such an application.

376 **Web Application, Server-Side**

377 The portion of a Relying Party application that executes server-side and re-
378 sponds to HTTP requests. When the term "Web Application" appears unqualified
379 or without specific context in FIDO documents, it generally refers to either the
380 client-side portion or the combination of both client-side and server-side pieces of
381 such an application.

382 Bibliography

383 *Non-normative*

384 **[ISOBiometrics]** “Project Editor”. Harmonized Biometric Vocabulary. ISO/IEC JTC 1.
385 15 November 2007. See

386 http://isotc.iso.org/livelink/livelink/fetch/2000/2122/327993/327973/654118/6687752/N_3004_JTC_1_SC_37_-_Harmonized_Biometric_Vocabulary_-_for_information.pdf?nodeid=6719683&vernum=0

389 **[NSTCBiometrics]** Author Unknown. Biometrics Glossary. National Science and Tech-
390 nology Council. 14 September 2006. See <http://biometrics.gov/Documents/Glossary.pdf>