REVIEW DRAFT

FIDO Technical Glossary

FIDO Alliance Review Draft 02 July 2018

This version:
https://fidoalliance.org/specs/fido-v2.0-rd-20180702/fido-glossary-v2.0-rd-20180702.html
Previous version:
https://fidoalliance.org/specs/fido-v2.0-id-20180227/fido-glossary-v2.0-id-20180227.html
Editor:
Rolf Lindemann, Nok Nok Labs, Inc.
Contributors:
Davit Baghdasaryan, Nok Nok Labs, Inc.
Brad Hill, PayPal
Jeff Hodges, PayPal

Abstract

This document defines all the strings and constants reserved by UAF protocols. The values defined in this document are referenced by various UAF specifications.

Status of This Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current FIDO Alliance publications and the latest revision of this technical report can be found in the FIDO Alliance specifications index at https://www.fidoalliance.org/specifications/.

This document was published by the FIDO Alliance as a Review Draft. This document is intended to become a FIDO Alliance Proposed Standard. If you wish to make comments regarding this document, please Contact Us. All comments are welcome.

This is a Review Draft Specification and is not intended to be a basis for any implementations as the Specification may change. Permission is hereby granted to use the Specification solely for the purpose of reviewing the Specification. No rights are granted to prepare derivative works of this Specification. Entities seeking permission to reproduce portions of this Specification for other uses must contact the FIDO Alliance to determine whether an appropriate license for such use is available.

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

THIS FIDO ALLIANCE SPECIFICATION IS PROVIDED “AS IS” AND WITHOUT ANY WARRANTY OF ANY KIND, INCLUDING, WITHOUT LIMITATION, ANY EXPRESS OR IMPLIED WARRANTY OF NON-INFRINGEMENT,

MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Table of Contents

1. Notation

Type names, attribute names and element names are written as `code`.

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

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

1.1 Key Words

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

2. Introduction

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

This document is not an exhaustive compendium of all FIDO technical terminology because the FIDO terminology is built upon existing terminology. Thus many terms that are commonly used within this context are not listed. They may be found in the glossaries/documents/specifications referenced in the bibliography. Terms defined here that are not attributed to other glossaries/documents/specifications are being defined here.

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

3. Definitions

AAID

Authenticator Attestation ID. See Attestation ID.

Application

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

Application Facet

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

Application Facet ID

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

- For Web applications, the facet id is the RFC6454 origin [RFC6454].
- For Android applications, the facet id is the URI android:apk-key-hash:*<hash-of-apk-signing-cert>*
- For iOS, the facet id is the URI ios:bundle-id:*<ios-bundle-id-of-app>*

AppID

The AppID is an identifier for a set of different Facets of a relying party's application. The AppID is a URL pointing

The AppID is an identifier for a set of different Facets of a relying party's application. The AppID is a URL pointing to the TrustedFacets, i.e. list of FacetIDs related to this AppID.

Attestation

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

Attestation Certificate

A public key certificate related to an Attestation Key.

Authenticator Attestation ID / AAID

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

Attestation [Public / Private] Key

A key used for FIDO Authenticator attestation.

Attestation Root Certificate

A root certificate explicitly trusted by the FIDO Alliance, to which Attestation Certificates chain to.

Authentication

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

Authentication Algorithm

The combination of signature and hash algorithms used for authenticator-to-relying party authentication.

Authentication Scheme

The combination of an Authentication Algorithm with a message syntax or framing that is used by an Authenticator when constructing a response.

Authenticator, Authnr

See FIDO Authenticator.

Authenticator, 1stF / First Factor

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

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

Examples of such authenticator is a biometric sensor or a PIN based verification. Authenticators which only verify presence, such as a physical button, or perform no verification at all, cannot act as a first-factor authenticator.

Authenticator, 2ndF / Second Factor

A FIDO Authenticator which acts only as a second factor. Second-factor authenticators always require a single key handle to be provided before responding to a Sign command. They might or might not have a user verification method. It is assumed that these authenticators may or may not have an internal matcher.

Authenticator Attestation

The process of communicating a cryptographic assertion to a relying party that a key presented during authenticator registration was created and protected by a genuine authenticator with verified characteristics.

Authenticator Metadata

Verified information about the characteristics of a certified authenticator, associated with an AAID and available from the FIDO Alliance. FIDO Servers are expected to have access to up-to-date metadata to be able to interact with a given authenticator.

Authenticator Policy

A JSON data structure that allows a relying party to communicate to a FIDO Client the capabilities or specific authenticators that are allowed or disallowed for use in a given operation.

ASM / Authenticator Specific Module

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

AV

ASM Version

Bound Authenticator

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

Certificate

An X.509v3 certificate defined by the profile specified in [RFC5280] and its successors.

Channel Binding

See: [RFC5056], [RFC5929] and [ChannelID]. A channel binding allows applications to establish that the two end-points of a secure channel at one network layer are the same as at a higher layer by binding authentication to the higher layer to the channel at the lower layer.

Client

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

Confused Deputy Problem

A confused deputy is a computer program that is innocently fooled by some other party into misusing its authority. It is a specific type of privilege escalation.

Correlation Handle

Any piece of information that may allow, in the context of FIDO protocols, implicit or explicit association and or attribution of multiple actions, believed by the user to be distinct and unrelated, back to a single unique entity. An example of a correlation handle outside of the FIDO context is a client certificate used in traditional TLS mutual authentication: because it sends the same data to multiple Relying Parties, they can therefore collude to uniquely identify and track the user across unrelated activities. [AnonTerminology]

Deregistration

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

Discovery

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

E(K,D)

Denotes the Encryption of data D with key K

ECDAA

Elliptic Curve based Direct Anonymous Attestation. ECDAA is an attestation scheme alternative to FIDO Basic Attestation. It is an improved Direct Anonymous Attestation scheme based on elliptic curves and bilinear pairings. Direct Anonymous Attestation schemes use individual private keys in the Authenticator while avoiding global

correlation handles. ECDAA provides significantly improved performance compared with the original DAA scheme. FIDO ECDAA [FIDOEcdaaAlgorithm] defines object encodings, pairing friendly curves etc. in order to lead to interoperable ECDAA implementations across different FIDO Servers and FIDO Authenticators.

ECDSA

Elliptic Curve Digital Signature Algorithm, as defined by ANSI X9.62 [ECDSA-ANSI].

Enrollment

The process of making a user known to an authenticator. This might be a biometric enrollment as defined in [ISOBiometrics] or involve processes such as taking ownership of, and setting a PIN or password for, a non-biometric cryptographic storage device. Enrollment may happen as part of a FIDO protocol ceremony, or it may happen outside of the FIDO context for multi-purpose authenticators.

Facet

See Application Facet

Facet ID

See Application Facet ID

FIDO Authenticator

An authentication entity that meets the FIDO Alliance's requirements and which has related metadata.

A FIDO Authenticator is responsible for user verification, and maintaining the cryptographic material required for the relying party authentication.

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

A FIDO Authenticator may be referred to as simply an authenticator or abbreviated as "authnr". Important distinctions in an authenticator's capabilities and user experience may be experienced depending on whether it is a roaming or bound authenticator, and whether it is a first-factor, or second-factor authenticator.

It is assumed by registration assertion schemes that the authenticator has exclusive control over the data being signed by the attestation key.

Authenticators specify in the Metadata Statement whether they have exclusive control over the data being signed by the Uauth key.

FIDO Client

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

- A software component implemented in a user agent (either web browser or native application).
- A standalone piece of software shared by several user agents. (web browsers or native applications).

FIDO Data / FIDO Information

Any information gathered or created as part of completing a FIDO transaction. This includes but is not limited to, biometric measurements of or reference data for the user and FIDO transaction history.

FIDO Server

Server software typically deployed in the relying party's infrastructure that meets UAF protocol server requirements.

FIDO UAF Client

See FIDO Client.

FIDO User Device

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

Key Identifier (KeyID)

The KeyID is an opaque identifier for a key registered by an authenticator with a FIDO Server, for first-factor authenticators. It is used in concert with an AAID to identify a particular authenticator that holds the necessary key. Thus key identifiers must be unique within the scope of an AAID.

One possible implementation is that the KeyID is the SHA256 hash of the KeyHandle managed by the ASM.

KeyHandle

A key container created by a FIDO Authenticator, containing a private key and (optionally) other data (such as Username). A key handle may be wrapped (encrypted with a key known only to the authenticator) or unwrapped. In the unwrapped form it is referred to as a *raw key handle*. Second-factor authenticators must retrieve their key handles from the relying party to function. First-factor authenticators manage the storage of their own key handles, either internally (for roaming authenticators) or via the associated ASM (for bound authenticators).

Key Registration

The process of securely establishing a key between FIDO Server and FIDO Authenticator.

KeyRegistrationData (KRD)

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

KHAccessToken

A secret value that acts as a guard for authenticator commands. KHAccessTokens are generated and provided by an ASM.

Matcher

A component of a FIDO Authenticator which is able to perform (local) user verification, e.g. biometric comparison [ISOBiometrics], PIN verification, etc.

Matcher Protections

The security mechanisms that an authenticator may use to protect the matcher component.

Persona

All relevant data stored in an authenticator (e.g. cryptographic keys) are related to a single "persona" (e.g. “business” or “personal” persona). Some administrative interface (not standardized by FIDO) provided by the authenticator may allow maintenance and switching of personas.

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

This mechanism may be used to provide an additional measure of privacy to the user, where the user wishes to use the same authenticator in multiple contexts, without allowing correlation via the authenticator across those contexts.

PersonaID

An identifier provided by an ASM, PersonaID is used to associate different registrations. It can be used to create virtual identities on a single authenticator, for example to differentiate “personal” and “business” accounts. PersonaIDs can be used to manage privacy settings on the authenticator.

Reference Data

A (biometric) reference data (also called template) is a digital reference of distinct characteristics that have been

extracted from a biometric sample. Biometric reference data is used during the biometric user verification process [ISOBiometrics]. Non-biometric reference data is used in conjunction with PIN-based user verification.

Registration

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

Registration Scheme

The registration scheme defines how the authentication key is being exchanged between the FIDO Server and the FIDO Authenticator.

Relying Party

A web site or other entity that uses a FIDO protocol to directly authenticate users (i.e., performs peer-entity authentication). Note that if FIDO is composed with federated identity management protocols (e.g., SAML, OpenID Connect, etc.), the identity provider will also be playing the role of a FIDO Relying Party.

Roaming Authenticator

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

1. Using only its own internal storage for registrations
2. Allowing registered keys to be employed without access control mechanisms at the API layer. (Roaming authenticators still may perform user verification.)

Compare to Bound Authenticator.

S(K, D)

Signing of data D with key K

Server Challenge

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

Sign Counter

A monotonically increasing counter maintained by the Authenticator. It is increased on every use of the UAuth.priv key. This value can be used by the FIDO Server to detect cloned authenticators.

SignedData

A SignedData object is created and returned by an authenticator as the result of the authenticator's Sign command. The to-be-signed data input to the signature operation is represented in the returned SignedData object as intact values or as hashed values. The SignedData object also contains general information about the authenticator and its mode, a nonce, information about authenticator-specific cryptographic algorithms, and a use counter. The SignedData object is signed using a relying party-specific UAuth.priv key.

Silent Authenticator

FIDO Authenticator that does not prompt the user or perform any user verification.

Step-up Authentication

An authentication which is performed on top of an already authenticated session.

Example: The user authenticates the session initially using a username and password, and the web site later requests a FIDO authentication on top of this authenticated session.

One reason for requesting step-up authenication could be a request for a high value resource.

FIDO U2F is always used as a step-up authentication. FIDO UAF could be used as step-up authentication, but it could also be used as an initial authentication mechanism.

Note: In general, there is no implication that the step-up authentication method itself is "stronger" than the initial authentication. Since the step-up authentication is performed on top of an existing authentication, the resulting combined authentication strength will increase most likely, but it will never decrease.

Template

See reference data.

Test of User Presence

See User Presence Check

TLS

Transport Layer Security

Token

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

Transaction Confirmation

An operation in the FIDO protocol that allows a relying party to request that a FIDO Client, and authenticator with the appropriate capabilities, display some information to the user, request that the user authenticate locally to their FIDO Authenticator to confirm the information, and provide proof-of-possession of previously registered key material and an attestation of the confirmation back to the relying party.

Transaction Confirmation Display

This is a feature of FIDO Authenticators able to show content of a message to a user, and protect the integrity of this message. It could be implemented using the GlobalPlatform specified TrustedUI [TEESecureDisplay].

TrustedFacets

The data structure holding a list of trusted FacetIDs. The AppID is used to retrieve this data structure.

TTEXT

Transaction Text, i.e. text to be confirmed in the case of transaction confirmation.

Type-length-value/tag-length-value (TLV)

A mechanism for encoding data such that the type, length and value of the data are given. Typically, the type and length data fields are of a fixed size. This format offers some advantages over other data encoding mechanisms, that make it suitable for some of the FIDO UAF protocols.

Universal Second Factor (U2F)

The FIDO protocol and family of authenticators which enable a cloud service to offer its users the options of using an easy–to–use, strongly–secure open standards–based second-factor device for authentication. The protocol relies on the server to know the (expected) user before triggering the authentication.

Universal Authentication Framework (UAF)

. The FIDO Protocol and family of authenticators which enable a service to offer its users flexible and interoperable authentication. This protocol allows triggering the authentication before the server knows the user.

UAF Client

See FIDO Client.

UAuth.pub / UAuth.priv / UAuth.key

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

UINT8

An 8 bit (1 byte) unsigned integer.

UINT16

A 16 bit (2 bytes) unsigned integer.

UINT32

A 32 bit (4 bytes) unsigned integer.

UPV

UAF Protocol Version

User

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

User Agent

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

User Presence Check

The User Presence check in the authenticator verifies that some user is present at the authenticator and agrees with a generic authentication operation.

User Verification

The process by which a FIDO Authenticator locally authorizes use of key material, for example through a touch, pin code, fingerprint match or other biometric.

User Verification Token

The user verification token is generated by Authenticator and handed to the ASM after successful user verification. Without having this token, the ASM cannot invoke special commands such as Register or Sign.

The lifecycle of the user verification token is managed by the authenticator. The concrete techniques for generating such a token and managing its lifecycle are vendor-specific and non-normative.

Username

A human-readable string identifying a user's account at a relying party.

Verification Factor

The specific means by which local user verification is accomplished. e.g. fingerprint, voiceprint, or PIN.

This is also known as modality.

Web Application, Client-Side

The portion of a relying party application built on the "Open Web Platform" which executes in the context of the user agent. When the term "Web Application" appears unqualified or without specific context in FIDO documents, it generally refers to either the client-side portion or the combination of both client-side and server-side pieces of such an application.

Web Application, Server-Side

The portion of a relying party application that executes on the web server, and responds to HTTP requests. When the term "Web Application" appears unqualified or without specific context in FIDO documents, it generally refers to either the client-side portion or the combination of both client-side and server-side pieces of such an application.

A. References

A.1 Normative references

[FIDOEcdaaAlgorithm]

R. Lindemann; J. Camenisch; M. Drijvers; A. Edgington; A. Lehmann; R. Urian. *FIDO ECDAA Algorithm*. Review Draft. URL: https://fidoalliance.org/specs/fido-v2.0-rd-20180702/fido-ecdaa-algorithm-v2.0-rd-20180702.html

[RFC2119]

S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*. March 1997. Best Current Practice. URL: https://tools.ietf.org/html/rfc2119

A.2 Informative references

[AnonTerminology]

A. Pfitzmann; M. Hansen. *Anonymity, Unlinkability, Unobservability, Pseudonymity, and Identity Management - A Consolidated Proposal for Terminology, Version 0.34*. August 2010. URL: http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.34.pdf

[ChannelID]

D. Balfanz. *Transport Layer Security (TLS) Channel IDs*. Work In Progress. URL:http://tools.ietf.org/html/draft-balfanz-tls-channelid

[ECDSA-ANSI]

Public Key Cryptography for the Financial Services Industry - Key Agreement and Key Transport Using Elliptic Curve Cryptography ANSI X9.63-2011 (R2017). 2017. URL: https://webstore.ansi.org/RecordDetail.aspx?sku=ANSI+X9.63-2011+(R2017)

[ISOBiometrics]

ISO/IEC 2382-37 Harmonized Biometric Vocabulary. 2017. URL: https://www.iso.org/obp/ui/#iso:std:iso-iec:2382:-37:ed-2:v1:en

[RFC5056]

N. Williams. *On the Use of Channel Bindings to Secure Channels (RFC 5056)* November 2007. URL: http://www.ietf.org/rfc/rfc5056.txt

[RFC5280]

D. Cooper; S. Santesson; S. Farrell; S.Boeyen; R. Housley; W. Polk. *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. May 2008. URL: http://www.ietf.org/rfc/rfc5280.txt

[RFC5929]

J. Altman; N. Williams; L. Zhu. *Channel Bindings for TLS (RFC 5929)*. July 2010. URL: http://www.ietf.org/rfc/rfc5929.txt

[RFC6454]

A. Barth. *The Web Origin Concept (RFC 6454)*. June 2011. URL: http://www.ietf.org/rfc/rfc6454.txt

[TEESecureDisplay]

GlobalPlatform Trusted User Interface API Specifications. URL: https://www.globalplatform.org/specifications.asp