FIDO Authenticator Metadata Requirements

FIDO Alliance 24 May 2017

This version:
https://fidoalliance.org/specs/fido-uaf-v1.0-fd-20170524/fido-auth-metadata-v1.0-fd-20170524.html
Editor:
Meagan Karlsson, FIDO Alliance

Abstract

This document supports the FIDO Authenticator Certification program.

The fields in the Authenticator Metadata will be the primary method of communicating Authenticator Certification status and details about implementations to Relying Parties (RPs).

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 Final Document. If you wish to make comments regarding this document, please Contact Us. All comments are welcome.

Implementation of certain elements of this Document 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 this Document 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 DOCUMENT 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.

This document was published by the FIDO Alliance as a . If you wish to make comments regarding this document, please Contact Us. All comments are welcome.

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

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 reflects the Metadata Requirements for Authenticator Certification.

Mandatory fields are required to be evaluated by the FIDO Security Secretariat (Level 1), or the FIDO Accredited Security Laboratory (Level 2+) and submitted to FIDO as part of the Certification Request. Submitted metadata will be verified to be an accurate representation of the implementation.

Submission of Metadata to the FIDO Metadata Service (MDS) is optional, and can be done after recieving FIDO Authenticator Certification. If Metadata is submitted to MDS, the elements marked herin as Mandatory must be submitted and must match the Metadata submitted to FIDO during Authenticator Certification.

Functional Metadata Fields

The following Functional Metadata Fields are Mandatory for Authenticator Certification.

Field	Section	Description
VerificationMethodDescriptor	3.4	A descriptor for a specific base user verification method as implemented by the authenticator. A base user verification method must be chosen from the list of those described in [FIDORegistry].
verificationMethodANDCombination	3.5	VerificationMethodANDCombinations must be non-empty. It is a list containing the base user verification methods which must be passed as part of a successful user verification.
AAID	4.1	The Authenticator Attestation ID. See [UAFProtocol] for the definition of the AAID structure. This field must be set if the authenticator implements FIDO UAF.
AAGUID	4.1	The Authenticator Attestation GUID. See [FIDOKeyAttestation] for the definition of the AAGUID structure. This field must be set if the authenticator implements FIDO 2.0.
attestationCertificateKeyIdentifiers	4.1	A list of the attestation certificate public key identifiers encoded as hex string. This value must be calculated according to method 1 for computing the keyIdentifier as defined in [RFC5280] section 4.2.1.2. The hex string must not contain any non-hex characters (e.g. spaces). All hex letters must be lower case. This field must be set if neither AAID nor AAGUID are set. Setting this field implies that the attestation certificate(s) are dedicated to a single authenticator model.
description	4.1	A human-readable short description of the Authenticator.
authenticatorVersion	4.1	Earliest (i.e. lowest) trustworthy authenticatorVersion meeting the requirements specified in this metadata statement. Adding new StatusReport entries with status `UPDATE_AVAILABLE` to the metadata TOC object [FIDOMetadataService] must also change this authenticatorVersion if the update fixes severe security issues, e.g. the ones reported by preceding StatusReport entries with status code `USER_VERIFICATION_BYPASS`, `ATTESTATION_KEY_COMPROMISE`, `USER_KEY_REMOTE_COMPROMISE`, `USER_KEY_PHYSICAL_COMPROMISE`,

Field	Section	REVOKED. Description
protocolFamily	4.1	The FIDO protocol family. The values "uaf", "u2f", and "fido2" are supported. If this field is missing, the assumed protocol family is "uaf".
upv	4.1	The FIDO unified protocol version(s) (related to the specific protocol family) supported by this authenticator. See [UAFProtocol] for the definition of the Version structure.
userVerificationDetails	4.1	A list of alternative VerificationMethodANDCombinations. Each of these entries is one alternative user verification method. Each of these alternative user verification methods might itself be an "AND" combination of multiple modalities. All effectively available alternative user verification methods must be properly specified here. A user verification method is considered effectively available if this method can be used to either: 1) enroll new verification reference data to one of the user verification methods, or 2) unlock the UAuth key directly after successful user verification.
attachmentHint	4.1	A 32-bit number representing the bit fields defined by the ATTACHMENT_HINT constants in the FIDO Registry of Predefined Values [FIDORegistry].
isSecondFactorOnly	4.1	Indicates if the authenticator is designed to be used only as a second factor, i.e. requiring some other authentication method as a first factor (e.g. username+password).
tcDisplay	4.1	A 16-bit number representing a combination of the bit flags defined by the TRANSACTION_CONFIRMATION_DISPLAY constants in the FIDO Registry of Predefined Values [FIDORegistry]. This value must be 0, if transaction confirmation is not supported by the authenticator.
tcDisplayContentType	4.1	Supported MIME content type [RFC2049] for the transaction confirmation display, such as text/plain or image/png. This value must be present if transaction confirmation is supported, i.e. tcDisplay is non-zero.

3. Security Metadata Fields

The following Security-related Metadata Fields are Mandatory for Authenticator Certification.

Field	Section	Description
CodeAccuracyDescriptor	3.1	The *CodeAccuracyDescriptor* describes the relevant accuracy/complexity aspects of passcode user verification methods.
PatternAccuracyDescriptor	3.3	The *PatternAccuracyDescriptor* describes relevant accuracy/complexity aspects in the case that a pattern is used as the user verification method.
EcdaaTrustAnchor	3.8	In the case of ECDAA attestation, the ECDAA-Issuer's trust anchor must be specified in this field.
attestationRootCertificate	3.8	In the case of ECDAA attestation, the ECDAA-Issuer's trust anchor must be specified in this field.
assertionScheme	4.1	The assertion scheme supported by the authenticator. Must be set to one of the enumerated strings defined in the FIDO UAF Registry of Predefined Values [UAFRegistry].
authenticationAlgorithm	4.1	The authentication algorithm supported by the authenticator. Must be set to one of the ALG_ constants defined in the FIDO Registry of Predefined Values [FIDORegistry]. This value must be non-zero.
publicKeyAlgAndEncoding	4.1	The public key format used by the authenticator during registration operations. Must be set to one of the ALG_KEY constants defined in the FIDO Registry of Predefined Values [FIDORegistry]. Because this information is not present in APIs related to authenticator discovery or policy, a FIDO server must be prepared to accept and process any and all key representations defined for any public key algorithm it supports. This

Field	Section	value must be non-zero. Description
attestationTypes	4.1	The supported attestation type(s). (e.g. `TAG_ATTESTATION_BASIC_FULL`) See UAF Registry for more information [UAFRegistry].
keyProtection	4.1	A 16-bit number representing the bit fields defined by the `KEY_PROTECTION` constants in the FIDO Registry of Predefined Values [FIDORegistry]. This value must be non-zero.
matcherProtection	4.1	A 16-bit number representing the bit fields defined by the `MATCHER_PROTECTION` constants in the FIDO Registry of Predefined Values [FIDORegistry]. This value must be non-zero.
isKeyRestricted	2.16	This entry is set to `true`, if the Uauth private key is restricted by the authenticator to only sign valid FIDO signature assertions. This entry is set to `false`, if the authenticator doesn't restrict the Uauth key to only sign valid FIDO signature assertion. In this case, the calling application could potentially get any hash value signed by the authenticator. If this field is missing, the assumed value is `isKeyRestricted=true`.
isFreshUserVerificationRequired		This entry is set to `true`, if Uauth key usage always requires a fresh user verification. If this field is missing, the assumed value is `isFreshUserVerificationRequired=true`. This entry is set to `false`, if the Uauth key can be used without requiring a fresh user verification, e.g. without any additional user interaction, if the user was verified a (potentially configurable) caching time ago. In the case of `isFreshUserVerificationRequired=false`, the FIDO server must verify the registration response and/or authentication response and verify that the (maximum) caching time (sometimes also called "`authTimeout`") is acceptable. This entry solely refers to the user verification. In the case of transaction confirmation, the authenticator must always ask the user to authorize the specific transaction.

4. Biometric Metadata Fields

Providing the biometry related Metadata Statement field (i.e. *BiometricAccuracyDescriptor*) [FIDOMetadataStatement] is not mandatory for passing FIDO Authenticator Certification.

Use of Metadata Service 1.1 Status Dictionary

SRWG recommends the use of the Status Dictionary to report the issue dates of Certifications within the array of status report entries. Default status to as “not FIDO Certified” and status is updated to include Certifications as they are achieved. Each Certification would have a separate entry.

New Authenticator Certification Fields

SRWG recommends the following fields to be added to MDS, and that they become Mandatory for Security Certification.

Field	Description
CertificationDescriptor	Describes the externally visible aspects of the Security Certification evaluation.
CertNumber	The Authenticator certificate number. This is a unique per-Security Certified implementation identifier.
AuthTestVersion	The version of the Authenticator Security Test Procedures the implementation is Certified to, e.g. v1.0.
cryptoStrength	A claimed level of the overall cryptographic security, intended to give a Relying Party or consumer some insight into the level of cryptographic security supported by the Authenticator. Each key used by the Authenticator has a specified Cryptographic Strength, and the `overallClaimedCryptographicStrength` is less than or equal to the smallest of these Cryptographic Strengths. If this field is absent it indicates an unknown claimed overall cryptographic strength. For L2+ certified Authenticators the claimed overall cryptographic strength must be known and specified.

Field	Description
operatingEnv	A description of the particular operating environment that is used for the Authenticator. These are specified in [FIDORestrictedOperatingEnv]. ISSUE 1 Why do we need this? Is this intended to be a textual description only?

A. References

A.1 Normative references

[FIDOKeyAttestation]
FIDO 2.0: Key attestation format. URL: https://fidoalliance.org/specs/fido-v2.0-ps-20150904/fido-key-attestation-v2.0-ps-20150904.html

[FIDOMetadataService]
R. Lindemann, B. Hill, D. Baghdasaryan, *FIDO Metadata Service v1.0*. FIDO Alliance Implementation Draft. URLs:
HTML: fido-metadata-service-v1.1-id-20170202.html
PDF: fido-metadata-service-v1.1-id-20170202.pdf

[FIDOMetadataStatement]
B. Hill, D. Baghdasaryan, J. Kemp, *FIDO Metadata Statements v1.0*. FIDO Alliance Implementation Draft. URLs:
HTML: fido-metadata-statements.html
PDF: fido-metadata-statements.pdf

[FIDORegistry]
R. Lindemann, D. Baghdasaryan, B. Hill, *FIDO Registry of Predefined Values*. FIDO Alliance Implementation Draft. URLs:
HTML: fido-registry-v1.1-id-20170202.html
PDF: fido-registry-v1.1-id-20170202.pdf

[FIDORestrictedOperatingEnv]
Laurence Lundblade; Meagan Karlsson. *FIDO Authenticator Allowed Restricted Operating Environments List*. August 2016. Draft. URL: https://github.com/fido-alliance/security-requirements/blob/master/fido-authenticator-allowed-restricted-operating-environments-list.html

[RFC2049]
N. Freed, N. Borenstein, *Multipurpose Internet Mail Extensions (MIME) Part Five: Conformance Criteria and Examples (RFC 2049)*, IETF, November 1996, URL: http://www.ietf.org/rfc/rfc2049.txt

[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

[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*, IETF, May 2008, URL: http://www.ietf.org/rfc/rfc5280.txt

[UAFProtocol]
R. Lindemann, D. Baghdasaryan, E. Tiffany, D. Balfanz, B. Hill, J. Hodges, *FIDO UAF Protocol Specification v1.0*. FIDO Alliance Proposed Standard. URLs:
HTML: fido-uaf-protocol-v1.1-id-20170202.html
PDF: fido-uaf-protocol-v1.1-id-20170202.pdf

[UAFRegistry]
R. Lindemann, D. Baghdasaryan, B. Hill, *FIDO UAF Registry of Predefined Values*. FIDO Alliance Proposed Standard. URLs:
HTML: fido-uaf-reg-v1.1-id-20170202.html
PDF: fido-uaf-reg-v1.1-id-20170202.pdf