FIDO 2.0: Overview

FIDO Alliance Review Draft 04 October 2017

This version:
https://fidoalliance.org/specs/fido-undefined-v2.0-rd-20171004/fido-overview-v2.0-rd-20171004.html
Editor:
Michael B. Jones, Microsoft

Abstract

The FIDO 2 proposals present a strong user authentication framework that can replace passwords and will achieve it without compromising user convenience and experiences across difference types of devices and clients. This framework replaces password with FIDO 2.0 credentials that can't be phished, replayed, and are not subject to server breach attacks. Users can conveniently use gestures such as the use of PINs or biometrics to authorize use of FIDO 2.0 credentials. This document is an overview of the set of FIDO 2 specifications and is a recommended first-read before reading the individual FIDO 2 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.

Table of Contents

1. Problem Statement and Goals

Today's authentication technologies have a number of well-known shortcomings when used at Internet scale. The authentication experience for a user is extremely fragmented and anxiety-inducing. The user must authenticate themselves multiple times to different entities, including their local devices and various online services such as email and banking. Each such experience is different, with different failure modes. Passwords, by far the most common authentication method, have a number of problems related to strength, management and compromise.

There is a growing consensus that the industry needs to evolve to a better authentication model. The FIDO 2 specifications define a strong authentication architecture with following goals in mind:

It must be non-phishable. The authentication method must resist phishing attacks.

Secure against server break-ins. Many password compromises today result from server-side attacks where a huge list of passwords or password hashes gets exposed.

Machine-bound authentication and authorization. It should not be possible to take identity and service tickets from one machine and use them on different machines. This raises the bar for malware, as the malware must now stay on the victim machine in order to use any captured tickets.

It must provide a cryptographic proof (i.e., attestation) of the nature of the credentials used.

It must improve usability compared to passwords.

It must allow for differentiation based on the client platform and the strength of authentication. Specifically, while all platforms must be able to reach a meaningful level of strength, it should be possible to support scenarios where an RP's policy requires a specific client platform or a higher authentication strength to enable the scenario.

Preserves privacy of the user: Users have various online identities. The solution must provide

isolation between those identities and not allow relying parties or identity providers to link them together.

It must require a user gesture to perform a user authentication operation.

The FIDO 2 Requirements document describes these requirements in detail.

2. Overview of FIDO 2 Specifications

A core use case is where a user will show a gesture on a mobile device (authenticator) wants to login to a platform device (i.e. laptop, tablet, etc.). Another core use case is where the user wants to use a built-in authenticator to login to the same device - i.e. the authenticator device and the host device are the same. Assume the login is on to a web page in defining APIs to enforce clarity.

To address these use cases we need (a) A web API for strong authentication and (b) an Inter-device Protocol for Strong Authentication Protocol which specifies what will flow between host and authenticator when the web API is called on the host.

With this background, read the following specifications in this order:

FIDO Web API - defines a web API that allows web pages to access strong cryptographic credentials through browser Javascript. In order to ease developer adoption, it aims to integrate well with existing web platforms by reusing existing web specifications where possible

FIDO Client To Authenticator Protocol - defines an application layer protocol for communication between a personal device with cryptographic capabilities and a host computer that wishes to use these capabilities for strong user authentication. This protocol can be run over a variety of transport protocols using different physical media. This specification defines requirements for such transport protocols, but does not specify the details of how such transport layer connections should be set up

FIDO Key Attestation - defines generic data structures that cover the semantics of FIDO attestation. The specification provides a profile of these structures when a TPM acts as a FIDO authenticator. More profiles are expected to be added as the specification evolves.

With a lead in like that, the email to UFS can say "Here is a propsoal for starting the technical discussion, read the overview first for a guide to the docs:

Unified FIDO Specifications (UFS) Overview

This document.

2.1 FIDO 2 Requirements

Defines the requirements to be achieved by the FIDO 2 specifications and deployments. A working draft is located at this location.

2.2 FIDO Glossary

Defines the terminology used by the FIDO 2 specifications. The definitions of the terms FIDO Authenticator, FIDO 2.0 credential, and User Gesture are already proposed in the FIDO Requirements specification.

2.3 FIDO Attestation

Defines attestation formats used to validate FIDO Authenticators, uses of FIDO 2.0 credentials, and associated User Verification Methods.

2.4 FIDO Client To Authenticator Protocol

Describes a device-to-device protocol for communication between a personal device with cryptographic capabilities and a host computer that wishes to use these capabilities for security functions including strong user authentication.

2.5 FIDO Web API

Defines how to use the WebCrypto APIs to allow web pages to access strong credentials through browser JavaScript. It attempts to follow the spirit of the WebCrypto specification, while also being easy to use for developers. Because it specifies browser functionality, in order for it to be widely adopted, some aspects of this functionality will need to be developed in the W3C.

3. Related Specifications

These specifications are related to FIDO 2 and are likely to be used in combination with FIDO 2.

3.1 Token Binding over HTTP

Defines a collection of mechanisms that allow HTTP servers to cryptographically bind authentication tokens (such as cookies and OAuth tokens) to a TLS connection. See http://tools.ietf.org/html/draft-balfanz-https-token-binding.

3.2 Token Binding Protocol

Enables client/server applications to create long-lived, uniquely identifiable TLS bindings spanning multiple TLS sessions and connections. Applications are then enabled to cryptographically bind security tokens to the TLS layer, preventing token export and replay attacks. See https://tools.ietf.org/html/draft-popov-token-binding.

3.3 Federation Protocol Profiles

These profiles will define how particular federation protocols can request and employ FIDO 2 authentication and Token Binding. An OpenID Connect FIDO profile is planned. Other profiles, such as a SAML 2.0 profile are also possible, if there is working group/marketplace demand for them.