UAF Architectural Overview

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

Editors:

Rob Philpott, RSA, the Security Division of EMC
Sampath Srinivas, Google
John Kemp, FIDO Alliance

Contributors:

The following members of the FIDO UAF Alliance Technical Working Group(TWG) have contributed to this document:

Infineon Technologies
Lenovo
Nok Nok Labs
PayPal
RSA, The Security Division of EMC
Synaptics

Abstract:

The FIDO UAF strong authentication framework enables online services and websites, whether on the open Internet or within enterprises, to transparently leverage native security features of end-user computing devices for strong user authentication and to reduce the problems associated with creating and remembering many online credentials. The FIDO UAF Reference Architecture describes the components, protocols, and interfaces that make up the FIDO UAF strong authentication ecosystem.

Status:

This Specification has been prepared by FIDO Alliance, Inc. **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 Introduction....................................4

1.1 Background....................................4

1.2 FIDO UAF Documentation Roadmap....................................5

1.3 FIDO UAF Goals....................................6

2 FIDO UAF High-Level Architecture....................................9

2.1 FIDO UAF Client....................................9

2.2 FIDO UAF Server....................................10

2.3 FIDO UAF Protocols....................................10

2.4 FIDO UAF Authenticator Abstraction Layer....................................11

2.5 FIDO UAF Authenticator 11

2.6 FIDO UAF Authenticator Metadata Validation....................................12

3 FIDO UAF Usage Scenarios and Protocol Message Flows....................................13

3.1 FIDO UAF Authenticator Acquisition and User Enrollment....................................13

3.2 Authenticator Registration....................................13

3.3 Authentication....................................14

3.4 Step-up Authentication....................................16

3.5 Secure Transaction Confirmation....................................16

3.6 Adoption of New Types of FIDO UAF Authenticators....................................17

4 Relationship to Other Technologies....................................19

4.1 OpenID, SAML, and OAuth 19

4.2 OATH, TCG, PKCS#11, and ISO 24727....................................20

1 Introduction

This document describes the FIDO Universal Authentication Framework (UAF) Reference Architecture. The target audience for this document is decision makers and technical architects who need a high-level understanding of the FIDO UAF strong authentication solution and its relationship to other relevant industry standards.

The FIDO UAF specifications are as follows:

1. **FIDO UAF Protocol**
2. **FIDO UAF Application API and Transport Binding**
3. **FIDO UAF Authenticator Commands**
4. **FIDO UAF Authenticator-Specific Module API**
5. **FIDO UAF Authenticator Metadata**
6. **FIDO Registry of Predefined Values**
7. **FIDO Security Reference**

A glossary of terms used in the FIDO specifications is also available:

8. **FIDO Glossary**

These documents may all be found on the FIDO Alliance website at http://fidoalliance.org/specifications/download/

1.1 Background

The FIDO Alliance mission is to change the nature of online strong authentication by:

- Developing technical specifications defining open, scalable, interoperable mechanisms that supplant reliance on passwords to securely authenticate users of online services.
- Operating industry programs to help ensure successful worldwide adoption of the specifications.
- Submitting mature technical specifications to recognized standards development organization(s) for formal standardization.

The core ideas driving the FIDO Alliance’s efforts are 1) ease of use, 2) privacy and security, and 3) standardization. The primary objective is to enable online services and websites, whether on the open Internet or within enterprises, to leverage native security features of end-user computing devices for strong user authentication and to reduce the problems associated with creating and remembering many online credentials.

There are two key protocols included in the FIDO architecture that cater to two basic options for user experience when dealing with Internet services. The two protocols share many of underpinnings but are tuned to the specific intended use cases.

Universal Authentication Framework (UAF) Protocol

The UAF protocol allows online services to offer password-less and multi-factor security. The user registers their device to the online service by selecting a local authentication mechanism such as swiping a finger, looking at the camera, speaking into the mic, entering a PIN, etc. The UAF protocol allows the service to select which mechanisms are presented to the user.

Once registered, the user simply repeats the local authentication action whenever they need to authenticate to the service. The user no longer needs to enter their password when authenticating from that device. UAF also allows experiences that combine multiple authentication mechanisms such as fingerprint + PIN.

This document that you are reading describes the UAF reference architecture.

Universal 2nd Factor (U2F) Protocol

The U2F protocol allows online services to augment the security of their existing password infrastructure by adding a strong second factor to user login. The user logs in with a username and password as before. The service can also prompt the user to present a second factor device at any time it chooses. The strong second factor allows the service to simplify its passwords (e.g. 4–digit PIN) without compromising security.

During registration and authentication, the user presents the second factor by simply pressing a button on a USB device or tapping over NFC. The user can use their FIDO U2F device across all online services that support the protocol leveraging built–in support in web browsers.

Please refer to the FIDO website for an overview and documentation set focused on the U2F protocol.

1.2 FIDO UAF Documentation Roadmap

To understand the FIDO UAF protocol, it is recommended that new audiences start by reading this architectural overview document they are currently reading and become familiar with the technical terminology used in the specifications (the glossary). Then they should proceed to the individual UAF documents in the recommended order listed below.

- FIDO UAF Overview: This document. Provides an introduction to the FIDO UAF architecture, protocols, and specifications.
- FIDO Technical Glossary: Defines the technical terms and phrases used in FIDO Alliance specifications and documents.
- Universal Authentication Framework (UAF)
  - UAF Protocol: Message formats and processing rules for all UAF protocol messages.

- ○ UAF Application API and Transport Binding Specification: APIs and interoperability profile for client applications to utilize FIDO UAF.
- ○ UAF Authenticator Commands: Low-level functionality that UAF Authenticators should implement to support the UAF protocol.
- ○ UAF Authenticator-specific Module API: Authenticator-specific Module API provided by an ASM to the FIDO client.
- ○ UAF Authenticator Metadata: Information describing form factors, characteristics, and capabilities of FIDO UAF Authenticators used to inform interactions with and make policy decisions about the authenticators.
- ○ UAF Registry of Predefined Values: defines all the strings and constants reserved by UAF protocols.

- FIDO Security Reference: Provides an analysis of FIDO security based on detailed analysis of security threats pertinent to the FIDO protocols based on its goals, assumptions, and inherent security measures.

The remainder of this Overview section of the reference architecture document introduces the key drivers, goals, and principles which inform the design of FIDO UAF.

Following the Overview, this document describes:

- A high-level look at the components, protocols, and API's defined by the architecture
- The main FIDO UAF use cases and the protocol message flows required to implement them.
- The relationship of the FIDO protocols to other relevant industry standards.

1.3 FIDO UAF Goals

In order to address today's strong authentication issues and develop a smoothly-functioning low-friction ecosystem, a comprehensive, open, multi-vendor solution architecture is needed that encompasses:

- User devices, whether personally acquired, enterprise-issued, or enterprise BYOD, and the device's potential operating environment, e.g. home, office, in the field, etc.
- Authenticators[1]
- Relying party applications and their deployment environments
- Meeting the needs of both end users and Relying Parties
- Strong focus on both browser- and native-app-based end-user experience

[1] Also known as: authentication tokens, security tokens, etc.

This solution architecture must feature:

- FIDO UAF Authenticator discovery, attestation, and provisioning
- Cross-platform strong authentication protocols leveraging FIDO UAF Authenticators
- A uniform cross-platform authenticator API
- Simple mechanisms for Relying Party integration

The FIDO alliance envisions an open, multi-vendor, cross-platform reference architecture with these goals:

- **Support strong, multi-factor authentication**: Protect Relying Parties against unauthorized access by supporting end user authentication using two or more strong authentication factors (“something you know”, “something you have”, “something you are”).
- **Build on, but not require, existing device capabilities**: Facilitate user authentication using built-in platform authenticators or capabilities (fingerprint sensors, cameras, microphones, embedded TPM hardware), but do not preclude the use of discrete additional authenticators.
- **Enable Selection of the authentication mechanism**: Facilitate Relying Party and user choice amongst supported authentication mechanisms in order to mitigate risks for their particular use cases.
- **Simplify integration of new authentication capabilities**: Enable organizations to expand their use of strong authentication to address new use cases, leverage new device’s capabilities, and address new risks with a single authentication approach.
- **Incorporate extensibility for future refinements and innovations**: Design extensible protocols and APIs in order to support the future emergence of additional types of authenticators, authentication methods, and authentication protocols, while maintaining reasonable backwards compatibility.
- **Leverage existing open standards where possible, openly innovate and extend where not**: An open, standardized, royalty-free specification suite will enable the establishment of a virtuous-circle ecosystem, and decrease the risk, complexity, and costs associated with deploying strong authentication. Existing gaps – notably uniform authenticator provisioning and attestation, a uniform cross-platform authenticator API, as well as a flexible strong authentication challenge-response protocol leveraging the user's authenticators – will be addressed..
- **Complement existing single sign-on, federation initiatives:** While industry initiatives (such as OpenID, OAuth, SAML, and others) have created mechanisms to reduce the reliance on passwords through single sign-on or federation technologies, they do not directly address the need for an initial strong authentication interaction between end users and Relying Parties.

- **Preserve the privacy of the end user:** Provide the user control over the sharing of device capability information with Relying Parties, and mitigate the potential for collusion amongst Relying Parties.
- **Unify end-User Experience:** Create easy, fun, and unified end-user experiences across all platforms and across similar Authenticators.

2 FIDO UAF High-Level Architecture

The FIDO UAF Reference Architecture is designed to meet the FIDO goals and yield the desired ecosystem benefits. It accomplishes this by filling in the status-quo's gaps using standardized protocols and APIs.

The following diagram summarizes the reference architecture and how its components relate to typical user devices and Relying Parties:

The FIDO-specific components of the reference architecture are described below.

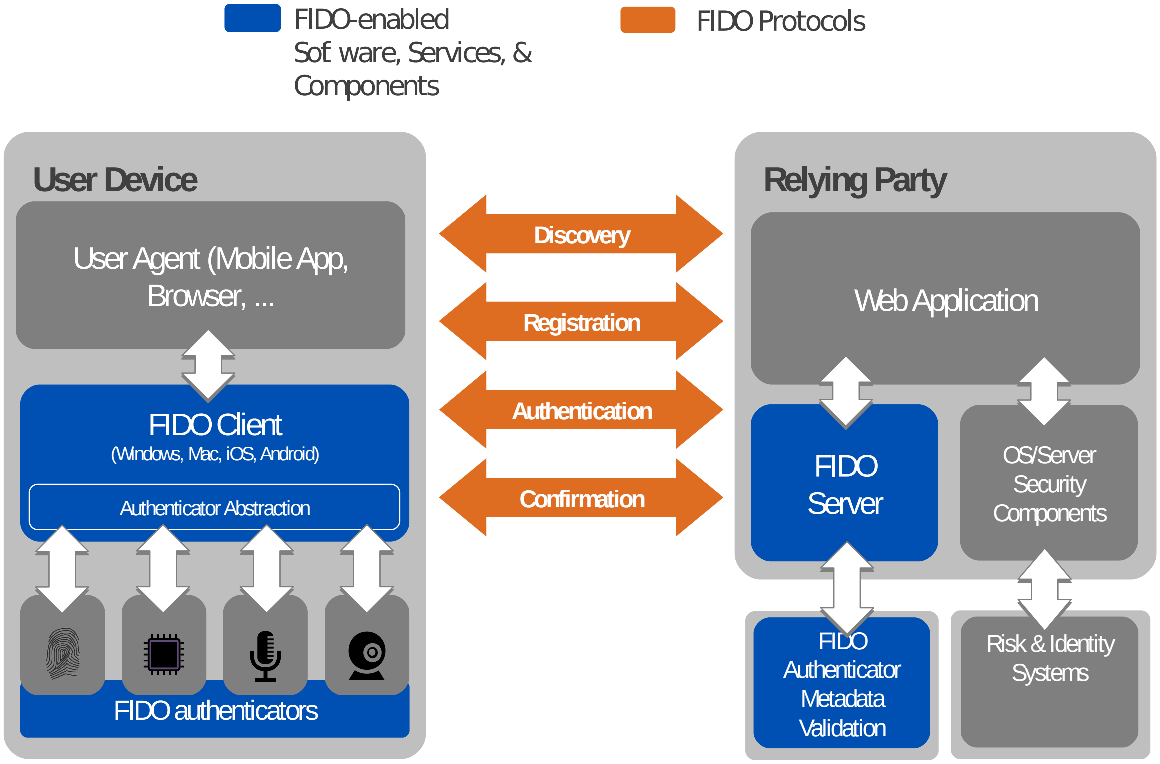
FIDO-enabled Software, Services, & Components
FIDO Protocols
User Device
User Agent (Mobile App, Browser, ...
FIDO Client
(Windows, Mac, iOS, Android)
Authenticator Abstraction
FIDO authenticators
Discovery
Registration
Authentication
Confirmation
Relying Party
Web Application
FIDO Server
OS/Server Security Components
FIDO Authenticator Metadata Validation
Risk & Identity Systems

Figure 2.1: FIDO UAF High-Level Architecture

2.1 FIDO UAF Client

A FIDO UAF Client implements the client side of the FIDO UAF protocols, and is responsible for:

- Interacting with specific FIDO UAF Authenticators using the FIDO UAF Authenticator Abstraction layer via the FIDO UAF Authenticator API.
- Interacting with a user agent on the device (e.g. a mobile app, browser) using user agent-specific interfaces to communicate with the FIDO UAF Server. For example, a FIDO-specific browser plugin would use existing browser plugin interfaces or a mobile app may use a FIDO-specific SDK. The user agent is then responsible for communicating FIDO UAF messages to a FIDO UAF Server at a Relying Party.

The FIDO UAF architecture ensures that FIDO client software can be implemented across a range of system types, operating systems, and Web browsers. While FIDO client software is typically platform-specific, the interactions between the components should ensure a consistent user experience from platform to platform.

2.2 FIDO UAF Server

A FIDO UAF server implements the server side of the FIDO UAF protocols and is responsible for:

- Interacting with the Relying Party web server to communicate FIDO UAF protocol messages to a FIDO UAF Client via a device user agent.
- Validating FIDO UAF authenticator attestations against the configured authenticator metadata to ensure only trusted authenticators are registered for use.
- Manage the association of registered FIDO UAF Authenticators to user accounts at the Relying Party.
- Evaluating user authentication and transaction confirmation responses to determine their validity.

The FIDO UAF server is conceived as being deployable as an on-premise server by Relying Parties or as being outsourced to a FIDO-enabled third-party service provider.

2.3 FIDO UAF Protocols

The FIDO UAF protocols carry FIDO UAF messages between user devices and Relying Parties. There are protocol messages addressing:

- Authenticator Registration: The FIDO UAF registration protocol enables Relying Parties to:
    - Discover the FIDO UAF Authenticators available on a user’s system or device. Discovery will convey FIDO UAF Authenticator attributes to the Relying Party thus enabling policy decisions and enforcement to take place.

  - Verify attestation assertions made by the FIDO UAF Authenticators to ensure the authenticator is authentic and trusted. Verification occurs using the attestation public key certificates distributed via authenticator metadata.
  - Register the authenticator and associate it with the user's account at the Relying Party. Once an authenticator attestation has been validated, the Relying Party can provide a unique secure identifier that is specific to the Relying Party and the FIDO UAF Authenticator. This identifier can be used in future interactions between the pair {RP, Authenticator} and is not known to any other devices.
- User Authentication: Authentication is typically based on cryptographic challenge-response authentication protocols and will facilitate user choice regarding which FIDO UAF Authenticators are employed in an authentication event.
- Secure Transaction Confirmation: If the user authenticator includes the capability to do so, a Relying Party can present the user with a secure message for confirmation. The message content is determined by the Relying Party and could be used in a variety of contexts such as confirming a financial transaction, a user agreement ,or releasing patient records.

2.4 FIDO UAF Authenticator Abstraction Layer

The FIDO UAF Authenticator Abstraction Layer provides a uniform API to FIDO Clients enabling the use of authenticator-based cryptographic services for FIDO-supported operations. It provides a uniform lower-layer “authenticator plugin” API facilitating the employment of multi-vendor FIDO UAF Authenticators and their requisite drivers.

2.5 FIDO UAF Authenticator

A FIDO UAF Authenticator is a secure entity, connected to or housed within FIDO user devices, that can create key material associated to a Relying Party. The key can then be used to participate in FIDO UAF strong authentication protocols. For example, the FIDO UAF Authenticator can provide a response to a cryptographic challenge using the key material thus authenticating itself to the Relying Party.

In order to meet the goal of simplifying integration of trusted authentication capabilities, a FIDO UAF Authenticator will be able to attest to its particular type (e.g., biometric) and capabilities (e.g., supported crypto algorithms), as well as to its provenance. This provides a Relying Party with a high degree of confidence that the user being authenticated is indeed the user that originally registered with the site.

2.6 FIDO UAF Authenticator Metadata Validation

In the FIDO UAF context, attestation is how Authenticators make claims to a Relying Party during registration that the keys they generate, and/or certain measurements they report, originate from genuine devices with certified characteristics. An attestation signature, carried in a FIDO UAF registration protocol message,is validated by the FIDO UAF Server. FIDO UAF Authenticators are created with attestation private keys used to create the signatures and the FIDO UAF Server validates the signature using that authenticator's attestation public key certificate located in the authenticator metadata. The metadata holding attestation certificates is shared with FIDO UAF Servers out of band.

3 FIDO UAF Usage Scenarios and Protocol Message Flows

The FIDO UAF ecosystem supports the use cases briefly described in this section.

3.1 FIDO UAF Authenticator Acquisition and User Enrollment

It is expected that users will acquire FIDO UAF Authenticators in various ways: they purchase a new system that comes with embedded FIDO UAF Authenticator capability; they purchase a device with an embedded FIDO UAF Authenticator, or they are given a FIDO Authenticator by their employer or some other institution such as their bank.

After receiving a FIDO UAF Authenticator, the user must go through an authenticator-specific *enrollment* process, which is outside the scope of the FIDO UAF protocols. For example, in the case of a fingerprint sensing authenticator, the user must register their fingerprint(s) with the authenticator. Once enrollment is complete, the FIDO UAF Authenticator is ready for *registration* with FIDO UAF enabled online services and websites.

3.2 Authenticator Registration

Given the FIDO UAF architecture, a Relying Party is able to transparently detect when a user begins interacting with them while possessing an initialized FIDO UAF Authenticator. In this initial introduction phase, the website will prompt the user regarding any detected FIDO UAF Authenticator(s), giving the user options regarding registering it with the website or not.

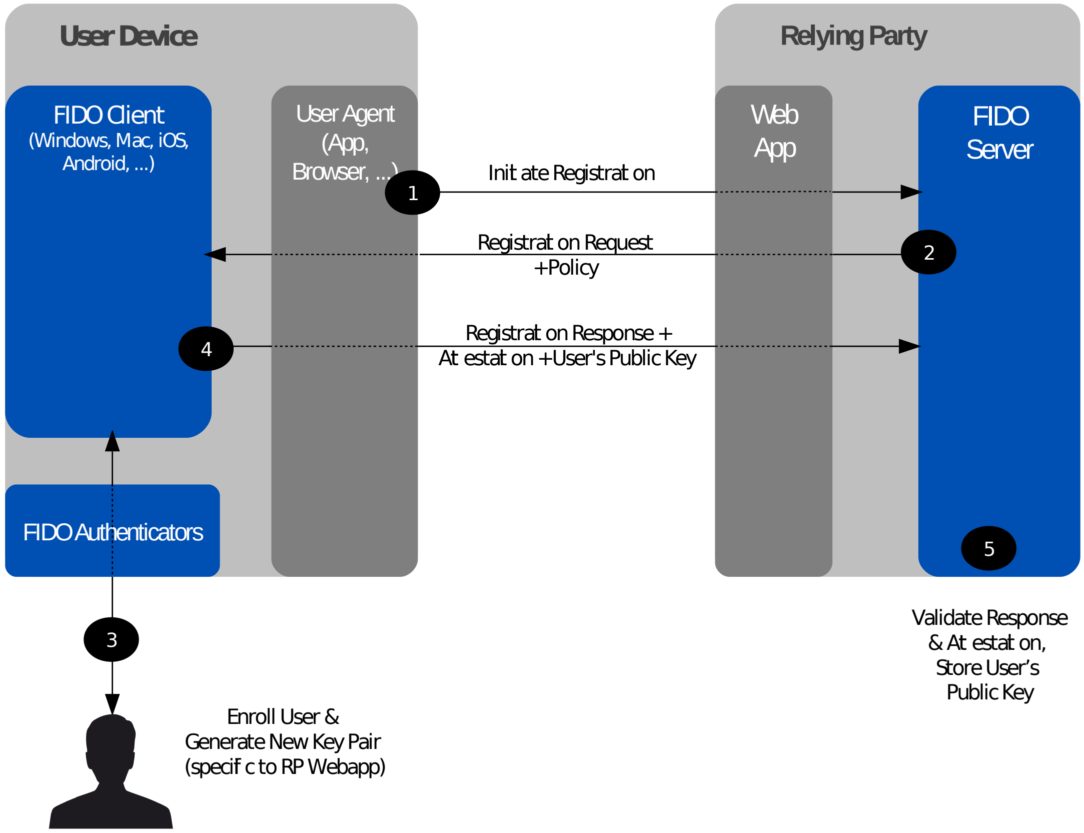
User Device
FIDO Client
(Windows, Mac, iOS, Android, ...)
User Agent
(App, Browser, ...)
Relying Party
Web App
FIDO Server
1
Initiate Registration
2
Registration Request +Policy
4
Registration Response + Attestation +User's Public Key
FIDO Authenticators
5
3
Validate Response & Attestation, Store User’s Public Key
Enroll User & Generate New Key Pair (specific to RP Webapp)

Figure 3.1: Registration Message Flow

3.3 Authentication

Following registration, the FIDO UAF Authenticator will be subsequently employed whenever the user authenticates with the website (and the authenticator is present). The website can implement various fallback strategies for those occasions when the FIDO Authenticator is not present. These might range from allowing conventional login with diminished privileges to disallowing login.

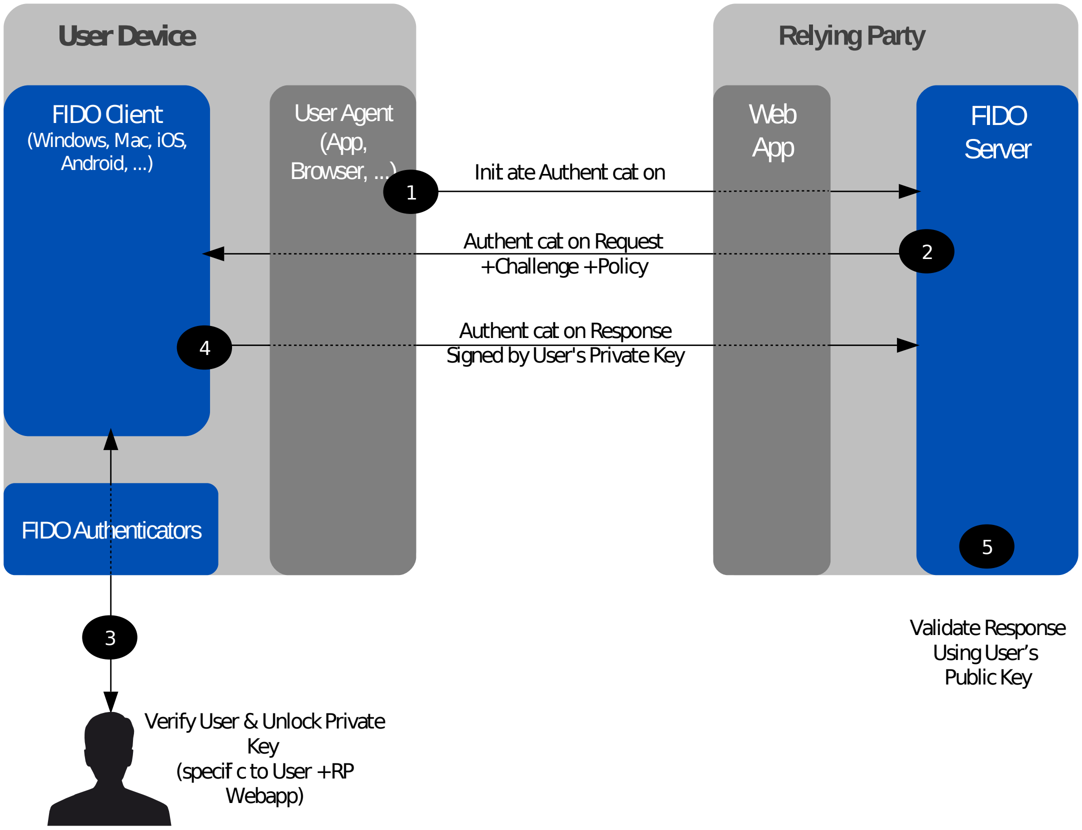

Figure 3.2: Authentication Message Flow

This overall scenario will vary slightly depending upon the type of FIDO UAF Authenticator being employed. Some authenticators may sample biometric data such as a face image, fingerprint, or voice print. Others will require a PIN or local authenticator-specific passphrase entry. Still others may simply be a hardware bearer authenticator. Note that it is permissible for a FIDO Client to interact with external services as part of the authentication of the user to the authenticator as long as the FIDO Privacy Principles are adhered to.

3.4 Step-up Authentication

Step-up authentication is an embellishment to the basic website login use case. Often, online services and websites allow unauthenticated, and/or only nominally authenticated use – for informational browsing, for example. However, once users request more valuable interactions, such as entering a members-only area, for example, the website may request further higher-assurance authentication. This could proceed in several steps, for example if the user then wishes to purchase something, with higher-assurance steps with increasing transaction value.

FIDO UAF will smoothly facilitate this interaction style since the website will be able to discover which FIDO UAF Authenticators are available on FIDO-wielding users' systems, and select incorporation of zero to all of them (or subsets thereof) in any particular authentication interaction. Thus online services and websites will be able to dynamically tailor initial, as well as step-up authentication interactions according to what the user is able to wield and the needed inputs to website's risk analysis engine given the interaction the user has requested.

3.5 Secure Transaction Confirmation

There are various innovative use cases possible given FIDO UAF-enabled Relying Parties with end-users wielding FIDO UAF Authenticators. Website login and step-up authentication are relatively simple examples. A somewhat more advanced use case is secure transaction processing.

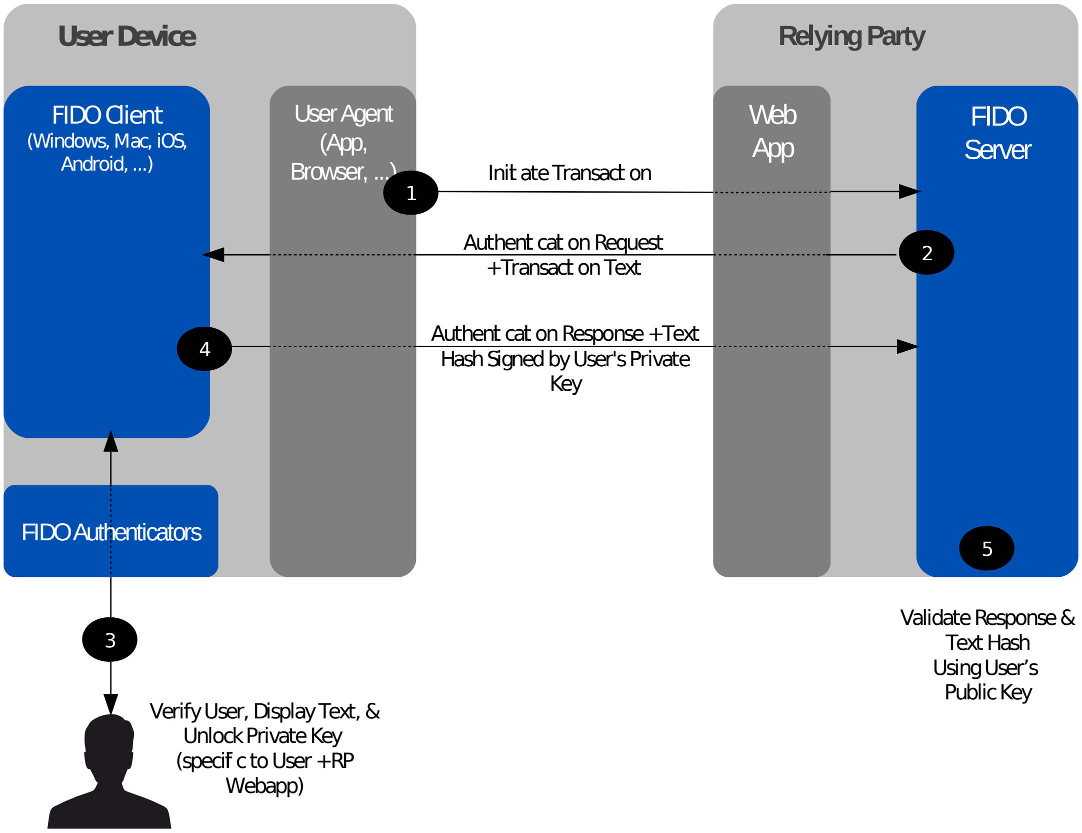
User Device
FIDO Client
(Windows, Mac, iOS, Android, ...)
User Agent (App, Browser, ...)
FIDO Authenticators
Relying Party
Web App
FIDO Server
1
Initiate Transaction
Authentication Request +Transaction Text
2
4
Authentication Response +Text Hash Signed by User's Private Key
3
Verify User, Display Text, & Unlock Private Key (specific to User +RP Webapp)
5
Validate Response & Text Hash Using User's Public Key

Figure 3.3: Confirmation Message Flow

Imagine a situation in which a Relying Party wants the end-user to confirm a transaction (e.g. financial operation, privileged operation, etc) so that any tampering of a transaction message during its route to the end device display and back can be detected. FIDO architecture has a concept of “secure transaction” which provides this capability. Basically if a FIDO UAF Authenticator has a secure display capability, FIDO UAF architecture makes sure that the system supports **What You See is What You Sign** mode (WYSIWYS). A number of different use cases can derive from this capability – mainly related to authorization of transactions (send money, perform a context specific privileged action, confirmation of email/address, etc).

3.6 Adoption of New Types of FIDO UAF Authenticators

Authenticators will evolve and new types are expected to appear in the future. Their adoption on the part of both users and Relying Parties is facilitated by the FIDO architecture. In order to support a new FIDO UAF Authenticator type, Relying Parties need only to add a new entry to their configuration describing the new authenticator, along with its FIDO Attestation Certificate. Afterwards, end users will be able to use the new FIDO UAF Authenticator type with those Relying Parties.

4 Relationship to Other Technologies

4.1 OpenID, SAML, and OAuth

FIDO protocols (both UAF and U2F) complement Federated Identity Management (FIM) frameworks, such as OpenID and SAML, as well as web authorization protocols, such as OAuth. FIM Relying Parties can leverage an initial authentication event at an identity provider (IdP). However, OpenID and SAML do not define specific mechanisms for direct user authentication at the IdP.

When an IdP is integrated with a FIDO-enabled authentication service, it can subsequently leverage the attributes of the strong authentication with its Relying Parties. The following diagram illustrates this relationship. FIDO-based authentication (1) would logically occur first, and the FIM protocols would then leverage that authentication event into single sign-on events between the identity provider and its federated Relying Parties (2).[2]

[2] FIM protocols typically convey IdP <-> RP interactions through the browser via HTTP redirects and POSTs.

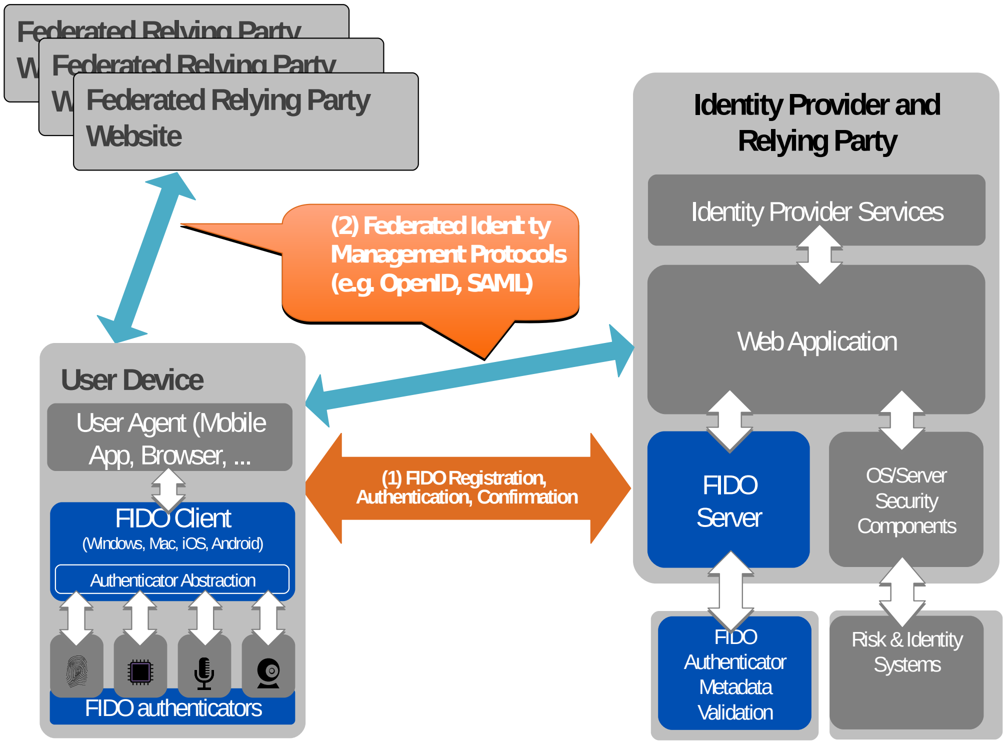
Federated Relying Party Website
Federated Relying Party Website
Federated Relying Party Website
(2) Federated Ident ty Management Protocols (e.g. OpenID, SAML)
Identity Provider and Relying Party
Identity Provider Services
Web Application
User Device
User Agent (Mobile App, Browser, ...
FIDO Client
(Windows, Mac, iOS, Android)
Authenticator Abstraction
FIDO authenticators
(1) FIDO Registration, Authentication, Confirmation
FIDO Server
OS/Server Security Components
FIDO Authenticator Metadata Validation
Risk & Identity Systems

Figure 4.1: FIDO UAF & Federated Identity Frameworks

4.2 OATH, TCG, PKCS#11, and ISO 24727

These are either initiatives (OATH, Trusted Computing Group (TCG)), or industry standards (PKCS#11, ISO 24727). They all share an underlying focus on hardware authenticators.

PKCS#11 and ISO 24727 define smart-card-based authenticator abstractions.

TCG produces specifications for the Trusted Platform Module, as well as networked trusted computing.

OATH, the "Initiative for Open AuTHentication", focuses on defining symmetric key provisioning protocols and authentication algorithms for hardware One-Time Password (OTP) authenticators.

The FIDO framework shares several core notions with the foregoing efforts, such as an authentication abstraction interface, authenticator attestation, key provisioning, and authentication algorithms. FIDO's work will leverage and extend some of these specifications.

Specifically, FIDO will complement them by addressing:

- Authenticator discovery
- User experience
- Harmonization of various authenticator types, such as biometric, OTP, simple presence, smart card, TPM, etc.