What Is FIDO2? The Open Standard Behind Passkeys, WebAuthn, and Phishing-Resistant Authentication

Definitions / Last updated 2026-04-27 / https://www.scrambleid.com/learn/what-is-fido2

In one sentence: FIDO2 is an open authentication standard, jointly developed by the FIDO Alliance and the W3C, that lets websites and applications use public-key cryptography (with the private key held in the user's hardware) to authenticate users without passwords and without phishing risk.

TL;DR (canonical)

- **FIDO2 = WebAuthn + CTAP.** WebAuthn is the W3C JavaScript API browsers expose to relying parties. CTAP is the FIDO Alliance protocol clients use to talk to external authenticators.
- **Phishing-resistant by construction.** The cryptographic assertion binds to the relying-party origin; a phishing site at a different origin cannot produce a valid assertion.
- **Hardware-bound private key.** The private key never leaves the user's secure hardware (Secure Enclave, TPM, Android Keystore, security key element).
- **Two authenticator form factors:** platform authenticators (Touch ID, Face ID, Windows Hello, Android biometric) built into the device, and roaming authenticators (security keys like YubiKey) that connect via USB, NFC, or Bluetooth.
- **Passkeys are FIDO2 credentials.** Specifically, discoverable credentials, often synced across devices.

The two specifications inside FIDO2

WebAuthn (W3C)

WebAuthn (Web Authentication) is the W3C JavaScript API. A relying-party website calls `navigator.credentials.create()` to register a new credential, or `navigator.credentials.get()` to authenticate an existing one. The browser:

1. Validates the origin of the calling page.
2. Constructs a challenge that includes the origin.
3. Asks the platform or roaming authenticator to sign the challenge.

4. Returns the signed assertion to the relying party.

The browser is doing the load-bearing work that makes FIDO2 phishing-resistant: the origin is part of the signed payload, and the browser will not let a script at one origin produce assertions for another.

CTAP (FIDO Alliance)

CTAP (Client to Authenticator Protocol) is the protocol that clients (browsers, OSes) use to communicate with external authenticators. CTAP2 is the current version. It defines:

- How the client requests a credential creation or authentication.
- How the authenticator returns the signed assertion.
- How user verification (PIN, biometric) is performed on the authenticator.
- Transport mechanisms: USB-HID, NFC, Bluetooth.

Platform authenticators do not technically use CTAP (they communicate through OS-level APIs), but they implement the same authenticator semantics.

How a FIDO2 ceremony works

Registration (creating a credential)

1. The user is logged in to the relying party (or in a self-service registration flow).
2. The relying party calls `navigator.credentials.create()` with parameters: relying-party ID (the domain), user account ID, supported algorithms, and authenticator preferences.
3. The browser asks the user to approve, typically with a biometric prompt or a tap on a security key.
4. The authenticator generates a fresh public-private key pair, stores the private key in hardware, and returns the public key plus an attestation.
5. The relying party stores the public key and credential ID against the user account.

The private key never leaves the authenticator. The public key is what the relying party stores.

Authentication (using a credential)

1. The user visits the relying party (logged out).
2. The relying party calls `navigator.credentials.get()` with a fresh challenge.
3. The browser asks the user to authenticate (biometric, PIN, or tap).
4. The authenticator signs the challenge plus the origin and other context.
5. The browser returns the signed assertion to the relying party.
6. The relying party verifies the signature using the stored public key.

Because the signed payload includes the origin (and the browser refuses to lie about origin), a phishing site cannot produce a valid assertion for the real origin.

The actors and terms you'll see

Term	What it means
Relying party (RP)	The website or service the user is authenticating to
Authenticator	The device that holds the private key (platform or roaming)
Client	The browser or platform that mediates between RP and authenticator
Platform authenticator	Built into the device (Touch ID, Face ID, Windows Hello, Android biometric)
Roaming authenticator / cross-platform authenticator	External device (security key) connecting via USB/NFC/Bluetooth
User verification (UV)	Local check that the authenticator's user is the right person (biometric, PIN)
User presence (UP)	Local check that a human is present (a tap or button press)
Resident key / discoverable credential	A credential stored on the authenticator with enough metadata to be looked up without the RP providing the credential ID first; passkeys are typically discoverable
Non-resident credential / non-discoverable credential	The RP must provide the credential ID; the credential is not enumerable from the authenticator alone
Attestation	A statement from the authenticator about its make/model/firmware, signed by a manufacturer key
Assertion	The signed authentication response returned to the RP

FIDO2 vs related concepts

Concept	What it is	Relationship to FIDO2
FIDO U2F	Original second-factor specification (2014)	Predecessor; both phishing-resistant; FIDO2 is more capable
FIDO UAF	Original passwordless specification for mobile	Predecessor; FIDO2 generalizes the model
WebAuthn	The W3C JavaScript API	One of the two specs that compose FIDO2
CTAP	The protocol to talk to authenticators	One of the two specs that compose FIDO2
Passkey	A discoverable FIDO2 credential, often synced	An implementation pattern using FIDO2
Phishing-resistant MFA	A property of an authentication ceremony	FIDO2 is one ceremony that satisfies this property
Passwordless	Authentication without a shared secret	FIDO2 is one of several passwordless approaches

Concept	What it is	Relationship to FIDO2
PKI smart card (PIV/CAC)	Federal/DoD credential	Different ceremony; both phishing-resistant; FIDO2 is more deployable on consumer devices

Where FIDO2 sits in the standards stack

```
Application code (relying party)
     |
     v
WebAuthn API (W3C)         ⟵ application calls navigator.credentials
     |
     v
Browser / OS                ⟵ mediates origin validation, user UI
     |
     v
CTAP (for roaming)          ⟵ protocol to external authenticator
     |
     v
Authenticator                ⟵ holds private key in hardware
     |
     v
Hardware-protected key store ⟵ Secure Enclave, TPM, security key chip
```

For platform authenticators, the CTAP layer is replaced by OS-level APIs, but the boundary between the application and the hardware-protected key store is the same.

What FIDO2 doesn't do

- **Identity proofing.** FIDO2 binds a cryptographic credential to an account; it does not establish that the person on the other end is who they claim to be at first registration. Pair FIDO2 with appropriate identity proofing (see What Is Identity Proofing?).
- **Authorization.** FIDO2 authenticates a user; what they're allowed to do is a separate decision (RBAC, ABAC, scoped tokens).
- **Recovery.** Lost authenticators need a recovery path. FIDO2 itself does not specify recovery; the relying party designs it. See Recovery and Fallback Playbook.
- **Cross-channel ceremonies.** FIDO2 is for browser-mediated authentication. For voice, in-person, or M2M channels, additional ceremonies are needed. See Omnichannel Authentication.

Key Takeaway

FIDO2 is an open authentication standard composed of two specifications, WebAuthn (W3C, the browser API) and CTAP (FIDO Alliance, the client-to-authenticator protocol). Together they enable phishing-resistant public-key cryptographic authentication where a hardware-protected private key signs a challenge that includes the relying-party origin. FIDO2 supports both platform authenticators (Touch ID, Face ID, Windows Hello, Android biometric) and roaming authenticators (security keys like YubiKey). Passkeys are an implementation of FIDO2 discoverable credentials. FIDO2 is the technical foundation of the phishing-resistant MFA movement (CISA, OMB M-22-09, FedRAMP) and is currently supported across all major browsers and operating systems.

FAQ

What is FIDO2?

FIDO2 is an authentication standard composed of two specifications: WebAuthn (the W3C API for browsers and platforms) and CTAP (the FIDO Alliance protocol for client-to-authenticator communication). Together they enable phishing-resistant public-key cryptographic authentication where a hardware-protected private key signs a challenge bound to the relying-party origin, and the public key is registered with the relying party.

What is the difference between FIDO2 and WebAuthn?

WebAuthn is the W3C specification for the JavaScript API that browsers and platforms expose to relying parties. FIDO2 is the umbrella term that includes WebAuthn plus CTAP (the protocol used to talk to external authenticators like security keys). When people say "FIDO2" they usually mean the full ecosystem; when they say "WebAuthn" they usually mean the browser-side API specifically.

What's the difference between FIDO U2F and FIDO2?

FIDO U2F is the predecessor specification, designed for second-factor use (you logged in with a password and added U2F as a second factor). FIDO2 expands the model to support primary authentication (replacing the password entirely), platform authenticators (Touch ID, Windows Hello, Android biometrics), passwordless and usernameless flows, and discoverable credentials (passkeys). Both are phishing-resistant; FIDO2 is the more capable and current standard.

Are passkeys the same as FIDO2?

Passkeys are an implementation of FIDO2 credentials, specifically discoverable credentials (resident keys) that are typically synced across the user's devices via their platform account (iCloud Keychain, Google Password Manager, etc.). Passkeys preserve FIDO2's phishing-resistance and are the consumer-facing brand for the technology. See What Are Passkeys?.

What is a platform authenticator vs a roaming authenticator?

A platform authenticator is built into the user's device (Touch ID, Face ID, Windows Hello, Android biometric) and uses the device's hardware-protected key store. A roaming authenticator (also called a cross-platform authenticator) is an external device the user plugs in or connects via NFC/Bluetooth, like a YubiKey. Both are FIDO2-compliant; both are phishing-resistant. The choice is about portability, recovery, and form factor.

Does FIDO2 require a hardware security key?

No. Platform authenticators built into modern devices (Touch ID, Face ID, Windows Hello, Android biometric) are FIDO2-compliant and use the hardware-protected key store on the device. A user with a recent iPhone, Android, Mac, or Windows machine likely already has a FIDO2 platform authenticator available. Hardware security keys are one form factor for FIDO2; not the only one.

References (public)

- W3C WebAuthn Level 2: https://www.w3.org/TR/webauthn-2/
- W3C WebAuthn Level 3 (current draft): https://www.w3.org/TR/webauthn-3/
- FIDO Alliance CTAP2 Specification: https://fidoalliance.org/specs/fido-v2.1-ps-20210615/fido-client-to-authenticator-protocol-v2.1-ps-20210615.html
- FIDO Alliance, Passkeys: https://fidoalliance.org/passkeys/
- CISA, Implementing Phishing-Resistant MFA: https://www.cisa.gov/sites/default/files/publications/fact-sheet-implementing-phishing-resistant-mfa-508c.pdf

---

Related reading

- What Is Phishing-Resistant MFA?
- What Are Passkeys?
- What Are NIST AAL Levels?
- Phishing-Resistant Web Authentication
- Recovery and Fallback Playbook