Què és WebAuthn?

WebAuthn (Web Authentication) és una API de navegador escrita en JavaScript que permet a les aplicacions web utilitzar mètodes d'autenticació forts i basats en criptografia de clau pública. Forma part d'un projecte més gran anomenat **FIDO2**, que és el fruit de la col·laboració entre l'aliança **FIDO** (Fast Identity Online) i el **W3C** (World Wide Web Consortium).

1. Els tres actors del sistema

Perquè WebAuthn funcioni, hi intervenen tres elements:

1. **El relying party (RP):** És el servidor (l'aplicació **CodeIgniter 4**). És qui "confia" en l'autenticació.
2. **El client (User Agent):** El navegador web (Chrome, Firefox, Safari). Actua d'intermediari i garanteix que la web que demana la clau és qui diu ser.
3. **L'authenticator:** El maquinari que genera les claus. Pot ser:
   - **Roaming:** Una clau USB (com una YubiKey).
   - **Platform:** El mateix dispositiu (el lector d'empremtes del MacBook, el FaceID de l'iPhone, el PIN de Windows Hello).

Els pilars del Standard

A. Criptografia de clau pública (Asimètrica)

A diferència de les contrasenyes o el TOTP, aquí no s'envia cap "secret" al servidor.

- Quan l'usuari es registra, l'**Authenticator** crea un parell de claus: una **pública** (que s'envia al servidor) i una **privada** (que no surt mai de l'Authenticator).
- L'autenticació es resol mitjançant un repte matemàtic: el servidor envia una dada, l'usuari la "signa" amb la clau privada, i el servidor verifica la signatura amb la clau pública.

B. Protecció contra el phishing (Origin-Bound)

Aquesta és la característica més potent. El navegador vincula la credencial al **domini (Origin)**.

> **Exemple:** Si un usuari rep un correu fals que el porta a `amazon-fals.com`, el navegador buscarà una clau per a aquest domini. Com que la clau real es va crear per a `amazon.com`, el navegador ni tan sols oferirà la possibilitat de signar. El phishing esdevé impossible.

C. Verificació de l'usuari (User verification)

L'estàndard permet que el servidor exigeixi una prova que l'usuari està present i és qui diu ser. Això es fa mitjançant l'autenticació local (biometria o PIN). El servidor rep un "flag" (bit) confirmant que la identitat de la persona s'ha verificat localment.

Flux de dades o protocol

Per explicar-ho a nivell de codi, WebAuthn defineix dos mètodes principals de l'API `navigator.credentials`:

1. `create()` **(Registre):** El servidor envia opcions (com el nom d'usuari i un challenge aleatori). El navegador genera la clau i retorna el "Attestation Object" (la clau pública).
2. `get()` **(Login/2FA):** El servidor envia el challenge. El navegador demana l'empremta/PIN, signa el challenge i retorna una "Assertion".

Per què s'anomena "passkey" ara?

Mentre que **WebAuthn** és el nom de l'estàndard tècnic, el terme **Passkey** és el nom comercial que Apple, Google i Microsoft han donat a les credencials WebAuthn que se sincronitzen al núvol (iCloud, Google Password Manager).

> **Diferència clau:** Una "Passkey" permet que si l'usuari crea la clau al seu iPhone, la pugui fer servir també al seu iPad o Mac sense haver de tornar-se a registrar, ja que la clau privada viatja de forma segura entre els seus dispositius.

Revisió núm. 1
Admin l’ha creat 2026-05-03 18:47:06 UTC
Admin l’ha actualitzat 2026-05-03 18:53:24 UTC