

PASSKEYS.

Phishing-resistant authentication built on **public-key cryptography**. The secret never leaves the user's device, never crosses the network, and cannot be replayed by an attacker who captures the traffic.

- W3C WEBAUTHN
- FIDO2 · CTAP2
- NO SHARED SECRET
- ORIGIN-BOUND

01 THE MECHANISM

Asymmetric key pair

The private key is generated inside the authenticator — secure enclave, TPM, or a hardware security key — and **never leaves it**. The relying party stores only the matching public key, so breaching the credential store yields nothing an attacker can replay.

Origin binding

The browser or platform — **not the user** — matches the requesting origin against the domain the credential was registered to. A look-alike domain or reverse-proxy relay has no credential to invoke, so the judgment call is taken away from the human.

Challenge-response

Every sign-in signs a **fresh, server-issued random challenge** together with the origin data. Nothing static crosses the wire, so a captured assertion is worthless the moment it is used — and a signature counter can flag cloning.

02 THE FLOW

Registration ONCE PER DEVICE

1

An existing verified session proves identity; the server issues a **random challenge** and its relying-party ID.

2

The device verifies the user locally — biometric or device PIN — and **mints a key pair scoped to that one domain**.

3

The public key and credential ID are returned and stored against the account; optional **attestation** proves the authenticator model.

Authentication EVERY SIGN-IN

1

The server sends a fresh challenge; the browser offers it **only to a credential registered for that exact origin**.

2

Local user verification unlocks the private key, which **signs the challenge together with the origin data**.

3

The server validates the signature against the stored public key — **possession plus inference**, cleared in a single gesture.

03 ATTACK SURFACE REMOVED

THREAT	WHY IT FAILS AGAINST A PASSKEY
Credential phishing	There is no shareable secret for the user to type, paste, or be talked out of.
Adversary-in-the-middle	Real-time proxy kits that defeat OTP and push MFA cannot forge the origin the signature is bound to.
Credential stuffing & reuse	Keys are unique per relying party and never user-chosen, so nothing carries across sites.
Replay & verifier breach	Signatures are single-use over a fresh challenge, and the stored public key is not a secret.
SIM swap & OTP theft	No out-of-band code, no shared seed, no help-desk-readable value anywhere in the path.

04 WHERE PASSKEYS SIT AGAINST THE ALTERNATIVES

METHOD	SHARED SECRET	PHISHING-RESISTANT	REPLAY-RESISTANT	TYPICAL AAL
Password	Yes	No	No	AAL1
SMS / email OTP	Yes	No	Partial	AAL2, restricted
TOTP authenticator	Yes (seed)	No	Partial	AAL2
Push approval	No	No — fatigue attacks	Partial	AAL2
Passkey	None	Yes — by construction	Yes	AAL2 · AAL3 hardware-bound

PASSKEYS.

OPERATE · GOVERN · COMMUNICATE

05

RECOVERY & LIFECYCLE

Enrolment is the new perimeter.

A passkey inherits the assurance of whatever proved identity at registration — bootstrap from a documented or in-person check, never from an emailed link.

Register at least two authenticators

per account so recovery is a second passkey rather than a fallback password. A surviving password path re-opens every threat above.

Synced vs. device-bound.

Synced credentials replicate through the platform account: recovery and adoption improve, but that account becomes an in-scope dependency. Device-bound keys stay in hardware and support attestation.

Revocation is server-side.

Deleting the public key ends the credential; joiner-mover-leaver and lost-device runbooks must enumerate credentials per account.

06

CONTROL MAPPING

NIST SP 800-63B

A multi-factor cryptographic authenticator meeting **AAL2**, and **AAL3** where the key is hardware-bound and verifier-impersonation-resistant. Synced passkeys qualify at AAL2 only.

PCI DSS V4.0

Satisfies **8.4** MFA for cardholder-data and administrative access, and **8.5.1**'s requirement that MFA systems resist replay.

PSD2 / SCA

Possession of the device-held key plus inherence or knowledge from local verification, with dynamic linking carried inside the signed challenge.

FIDO2 · WEBAUTHN

Open, vendor-neutral standards; attestation supports authenticator allow-listing under enterprise policy.

07

RESIDUAL RISK — WHAT PASSKEYS DO NOT SOLVE

Session theft

Tokens stolen **after** sign-in still work. Bind sessions to the device and shorten lifetimes.

Endpoint compromise

Malware on a verified device acts **as the user**. Device health signals stay in scope.

Help-desk recovery

Social engineering moves to the **reset path**. Harden and log identity re-proofing.

Sync account takeover

For synced credentials, the platform account inherits the trust. **Treat it as tier-0.**

08

MYTHS VS. REALITY

MYTH "My fingerprint or face gets sent to the website."	REALITY Biometrics never leave the device — they only unlock the local private key. The service receives a signature, never biometric data.
MYTH "If I lose my phone, I'm locked out for good."	REALITY Synced passkeys restore to a new device through the platform account — and a second registered authenticator means recovery never needs a password.
MYTH "A passkey is just a password stored somewhere else."	REALITY There is no shared secret anywhere . The server holds only a public key — worthless to steal, impossible to phish, never re-used across sites.
MYTH "Whoever steals my device can sign in as me."	REALITY The private key stays sealed behind local biometric or PIN verification . Possession of the hardware alone is not enough.
MYTH "It's proprietary vendor lock-in."	REALITY Passkeys are built on open W3C and FIDO standards — they work across platforms, browsers, password managers, and hardware keys.

09

SPEAK THE PROTOCOL — KEY TERMS

RELYING PARTY (RP)

The service being signed in to, identified by its **RP ID** — the domain a credential is permanently scoped to.

AUTHENTICATOR

What guards the private key: a **secure enclave**, **TPM**, **hardware security key**, or password manager.

ASSERTION

The signed package returned at each sign-in — **challenge plus origin data** — verified against the stored public key.

DISCOVERABLE CREDENTIAL

A passkey the authenticator can surface **without a username**, enabling one-gesture, usernameless sign-in.

ATTESTATION

Cryptographic proof of the **authenticator's make and model** at registration — the basis for enterprise allow-listing.

CTAP2

The FIDO protocol a browser uses to reach an **external authenticator** over USB, NFC, or Bluetooth.