HYPR Info Brief

Guide to Passkeys for Customer IAM Deployments

What Are Passkeys?

Passkeys remove the need for passwords, making the login experience easier and more secure. Passkey authentication eliminates credential stuffing and other attacks that use stolen or cracked passwords. It also protects users against phishing sites as the passkey is linked to a specific website or application.

Passkey authentication requires either biometric authentication, such as a fingerprint or facial recognition, a PIN or a swipe pattern (Android devices) for access. Passkeys are secret keys that stay on users' personal devices, and that can be used for authenticating to applications and websites. Passkeys leverage the Web Authentication API security standard, which uses public key cryptography for access. Each key is unique and created with encrypted data for added security.

Types of FIDO Passkeys

There are two primary types of passkeys, which differ in their usage and functional purpose.

Synced Passkeys

For the most part, this guide concerns the type of synced passkeys meant for consumer, not enterprise use. These are provided by and synced by the platform providers (i.e., Google, Microsoft and Apple). They have some security and functionality challenges — sharing introduces potential security issues, they don't meet regulatory requirements for an independent possession factor, they cannot be used for desktop login, and lack other critical enterprise capabilities — but the ability to sync across same platform devices makes them more consumer friendly.

Device-Bound Passkeys

A device-bound passkey cannot be passed amongst devices. It is designed for enterprise environments with security and operational requirements that make synced passkeys unsuitable. HYPR Enterprise Passkeys are built on this type of passkey. It operates within a technology stack, covering the entire range of enterprise use cases, from desktop to cloud. You can read more about the HYPR technology here.

App-Level Passkeys

App-level passkeys are similar to a single, device-bound passkey but dedicated to a specific mobile app; mobile browser capability is unavailable with this type of credential. These passkeys are not provided or managed by the platform providers (Google, Microsoft, Apple). They are useful for high value transactions but UX is limited.

Overview of How Passkeys Work

Passkeys work through an API called Web Authentication, commonly called WebAuthn. WebAuthn is a joint initiative of the World Wide Web Consortium (W3C) and the FIDO Alliance, an industry association that works to end our reliance on passwords.

Instead of a password, WebAuth uses public and private keys — otherwise known as public-key cryptography — to verify that the user is who they claim to be. Public and private keys are mathematically linked to one another. You can think of them like interlocking puzzle pieces; they're designed to go together, and you need both pieces to authenticate successfully. Unlike a traditional password, the private key is never shared with the site that the user is signing in to, or stored on an organization's servers.

When a user visits a website or application that supports passkeys, they can choose to secure their account with a passkey, rather than a traditional password. The website's server communicates information about the site, and asks the user to confirm their authenticator. This could be a phone, tablet or PC. A passkey — which includes a public and private key pair — gets generated for that specific website. This happens locally, on the user's device. The public key is stored on the website's server, while the private key is stored locally on the device. For individual/consumer passkeys, the private key can be passed to other devices using iCloud keychain or Google password manager.

When a user wants to sign into a website or application, the site sends an assertion challenge to the authenticator. The user must take the required action on the authenticator, such as entering a PIN or presenting a biometric. The authenticator then signs the authentication assertion with the user's private key and sends it back to the website. The website verifies the signed authentication assertion using its copy of the user's trusted public key, and they gain access.

All of this happens behind the scenes. All users need to worry about is setting up their passkey and unlocking it with the required action.

Introducing Passkeys to Your Users

By supporting the use of passkeys, you can offer your customers a login experience that is much more secure than passwords and MFA methods, such as SMS. Like any major change, however, you can take steps to minimize disruption and ease the transition for your customers and your business.

- Demonstrate to your targeted populations how it is not only easier than the typical password debacle we are now used to, but it is also more secure.
- Communicate the plan and timelines for critical dates.
- Deliver the upgrade message. Make it PR worthy!
- Define the order of roll outs to what populations and in what order.
- Monitor passkey usage and help requests.
- Consider making passkeys your primary method of authentication. Remember that most people are already leveraging biometrics on their smartphones. For many this will be an extension of that and a welcome change.

Who Creates Passkeys?

Synced passkeys are created by the user on their device and copied across to their Google, Apple, and Microsoft accounts on other connected phones, tablets, and laptops.

- Apple announced support in iOS 16 in Sep 2022, and iPadOS 16 and macOS Ventura in October 2022.
- Google announced support in Android starting October 2022 and ChromeOS in December 2022.
- Microsoft Windows is set to deliver support in 2023.

Many browsers already support sign-in with a passkey from a nearby device such as a mobile phone or security key. These include:

- Microsoft Edge and Google Chrome on Windows
- Edge, Safari and Google Chrome on macOS
- ChromeOS

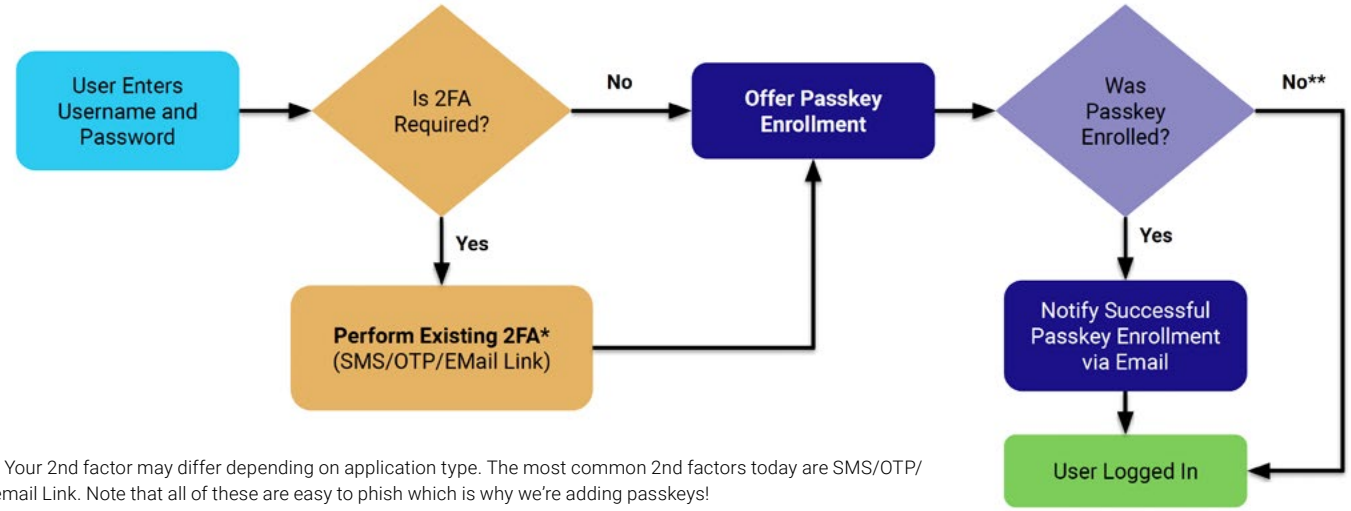
Enabling Synced Passkeys for Your Website or Mobile App

To enable passkeys for your website, you'll need the following:

- FIDO2 Certified server that supports passkeys
- Underlying platform supporting passkey syncing (iOS 16+, macOS Ventura, Android 12+)
- Front end web code to support passkey login - `navigator.credentials` Javascript API
- Mobile code to support passkey login, in the case of a mobile application
- The native mobile application will require a new enrollment or registration flow to be added

Onboarding Flow for Registering Customers With Passkeys

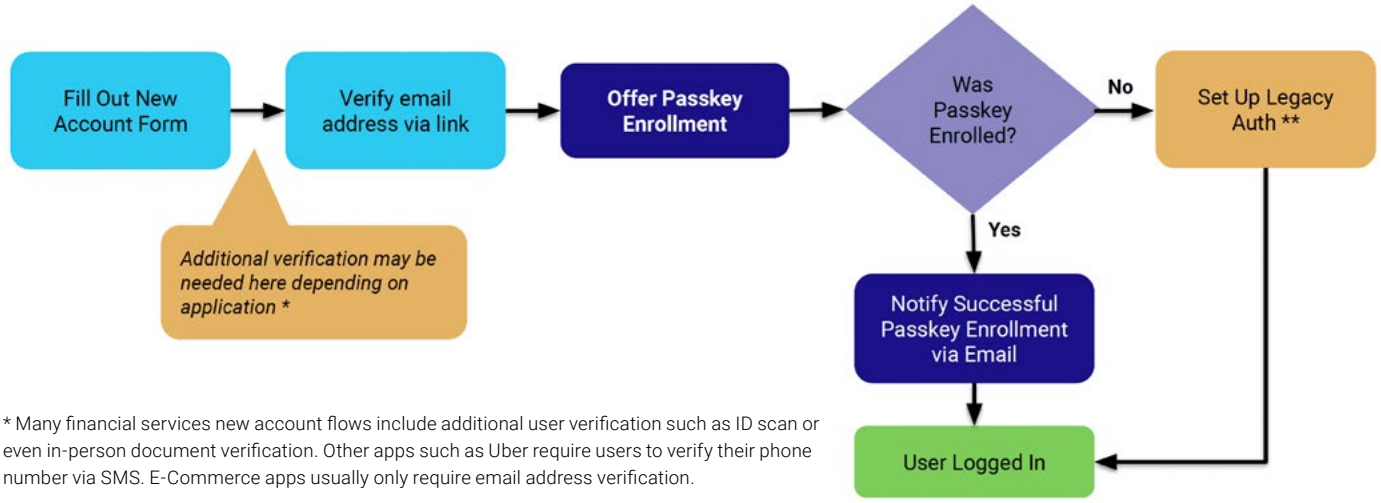
A good approach for onboarding customers with passkeys is to introduce passkey registration into your existing account sign-up flow as an additional authentication option. Your login page should also offer existing users the option to add passkeys as an alternative authentication method. If your application has an Account Management page, it should offer the option to manage (create/delete) passkeys.



* Your 2nd factor may differ depending on application type. The most common 2nd factors today are SMS/OTP/email Link. Note that all of these are easy to phish which is why we're adding passkeys!

** If a user does not enroll a passkey, it is helpful to show them a message explaining the value of passkeys.

Enrolling Synced Passkey for Existing User Account



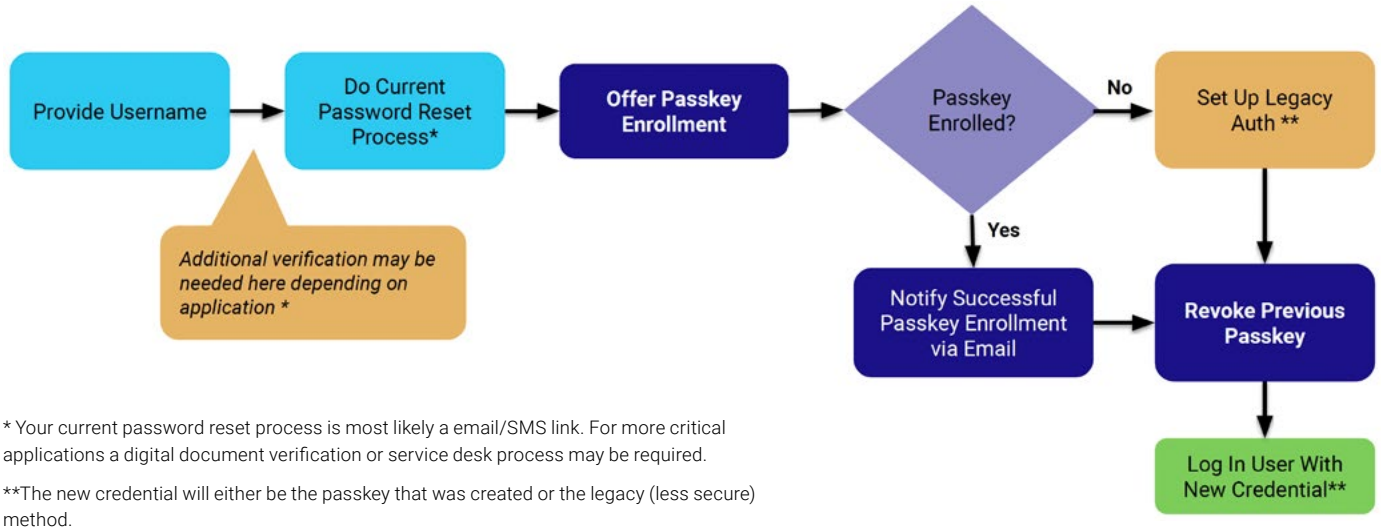
* Many financial services new account flows include additional user verification such as ID scan or even in-person document verification. Other apps such as Uber require users to verify their phone number via SMS. E-Commerce apps usually only require email address verification.

**This would most likely result in the user setting up a password or other legacy and easily phishable authentication factor. It is recommended that once a userenrolls a passkey, that this option goes away.

Enrolling Synced Passkey for New User Account

What If a Customer Loses or Deletes Their Passkey?

Depending on each particular scenario, there may be failsafes built-in to address this scenario. For example, Apple has an escrow account. These types of failsafes are dependent on how the account is set up. There are rare instances however, that the passkeys may be unrecoverable.



* Your current password reset process is most likely a email/SMS link. For more critical applications a digital document verification or service desk process may be required.

**The new credential will either be the passkey that was created or the legacy (less secure) method.

Dealing With Lost Passkeys (Synced)

How to Tell Which Customers Are Using Passkeys

Use the API to obtain detailed information about passkey usage.

What About Customers Without Mobile Phones?

Your customers without mobile phones can use passkeys but only on computers that support passkeys. In the case of Windows, the passkey is created locally on the operating system leveraging the Windows Hello PIN/biometric. The user can then log into the web applications on that workstation with the same passkey.

In order to create a passkey on Apple computers, iCloud Keychain must be set up on the user's Mac. Their Mac will also need to have a TouchID biometric fingerprint reader (e.g. MacBook or Magic Keyboard have TouchID). The passkey is stored in the user's iCloud Keychain and can be utilized on other Apple devices on the same iCloud account.

ADA Compliance

There are ancillary technologies that make passkeys available for the visually impaired. This area is expected to expand with more options coming to market as passkeys adoption grows.

Passkeys Security FAQs

Here we answer some of the most common questions we hear about the security implications of passkeys.

How is a passkey more secure than a password?

Passkeys are more secure and easier to use than the traditional password. Passkeys are a standard-based technology that, unlike passwords, are resistant to phishing, are always strong, and are designed so that there are no shared secrets. Passkeys cannot be written down or typed in which makes it harder for hackers to trick users into providing their authentication credentials. Passkeys are built on top of a broadly supported industry standard, the W3C Web Authentication API or WebAuthn, created by the World Wide Web Consortium and the FIDO Alliance, a group that has spent years developing approaches to reduce the effectiveness of phishing, eliminate hijacking, and increase authentication simplicity for users.

Are passkeys a form of multi-factor authentication?

Passkeys are kept on a user's devices (possession, i.e., something the user "has") and — if the relaying party (RP) requests User Verification — can only be exercised by the user with a biometric or PIN (inherence, i.e., something the user "is" or "knows"). Thus, authentication with passkeys embodies the key components of multi-factor security. However, because passkeys can be passed across devices, they do not satisfy the possession requirement under some regulations and are not currently recognized as an official form of multi-factor authentication under these regimes.[1]

Relying Parties may be concerned that a passkey could be made available to an attacker through a single factor (say, a password) from the platform vendor account. In practice, however, this is not usually the case: platform vendors consider multiple signals beyond the user's password — some visible to the user, some not — when authenticating users and restoring passkeys to their devices.

1 https://fidoalliance.org/passkeys/#faq

What are Bring Your Own Device (BYOD) security considerations of synced passkeys?

Synced passkeys in their current state, can be copied to other devices and shared. The platform vendors have made it extremely easy for end user adoption of passkeys but increase the security concerns for IT and Security Teams. Synced passkeys do offer better protection than passwords. It will require the IT and security team to accept the following concerns based on the current implementation state of passkeys:

- Passkeys can be shared through AirDrop on Apple devices
- Passkeys are replicated to every device the user has sync'ed to their iCloud account. There could be shared devices with a spouse, children and family friends
- Does not meet the current requirements for passwordless MFA
- The innate nature of FIDO2 privacy directives within the specification. See the specification 14.5.2. Authentication Ceremony Privacy

Synced passkeys are copied to all your devices that have your Apple, Google, and Microsoft accounts so that means if your corporate SSO portal uses passkeys, these credentials will be copied to these devices.

Do Apple, Google, Microsoft, etc. have access to my users' passkeys?

Synced passkeys are ultimately held on the provider's cloud. Theoretically, only the user can decrypt and access encrypted content.

Apple: Content is encrypted with an elliptical key (using P-256) derived from the user's iCloud account password.

Google: Password manager provides backup and sync. End-to-end encryption is performed using a key on the user's device.

How are passkeys copied across my users' devices securely?

It depends on the platform's implementation. Apple, for example, leverages their iCloud services to replicate and copy data across all user devices enrolled and configured for iCloud. Specifically the iCloud Passwords and Keychain capability. See Apple's documentation for more information.

Android and Chrome leverage the Google Password Manager. This service follows the same model as Apple. With one slight variation. The Google Password Manager is available for only Android and Chrome where Apple has access to its wider array of platform devices. See Google's documentation for more information.

Both platforms follow the same encryption techniques in which each device generates a unique key pair for the device as part of the encryption process. This removes the ability for the platform vendors to decrypt the data which resides within their data centers.

More on Passkeys

This guide discusses deployment of synced, consumer-level passkeys for your customers. Synced passkeys can pose multiple security and operational challenges for organizations, particularly in regulated industries. Authentication platforms that support both synced and device-bound passkeys provide a more secure, comprehensive solution that can address both workforce and customer populations. For information about HYPR's FIDO Certified passwordless authentication solution, and how it leverages synced passkeys and device-bound Enterprise Passkeys, contact a HYPR expert.

For information about passkeys from a user perspective, see A User's Guide to Passkeys.