

Testing Immediate Mediation on Chrome

This document is publicly visible.

Last updated on Jul 2, 2025 . The last edit added information about the Chrome Origin Trial.

These are instructions on how to try out the Immediate Mediation (single-click sign-in on Chrome) currently available on Chrome Canary.

Immediate Mediation is [a proposal](#) that has been put forward at the W3C WebAuthn Working Group. The proposal and the prototype are subject to change.

Origin Trial

The feature is open for [Origin Trial](#) registration on desktop Chrome 139, Web developers can sign up for the trial on [the Chrome Origin Trial console](#). Chrome 139 is currently on Chrome Beta channel and will be promoted to Stable in early August.

Configuration for Local Testing

To use this on a site that is not registered for the Origin Trial:

1. Download and open the latest version of [Chrome Canary on desktop](#).
2. Enable the Immediate Mediation flag in `chrome://flags`. You can just paste `chrome://flags/#web-authentication-immediate-get` into the address bar.
3. Be able to create usable passkeys and passwords for the test site you are using:
 - a. Passwords have to be saved in Google Password Manager.
 - b. Passkeys can be in Google Password Manager, in which case you also need to be signed in and syncing in Chrome with a Google account. You may also test with saved passkeys in Windows Hello or iCloud keychain.

Feature Detection

When the above flag is enabled on Chrome Canary, the availability of Immediate Mediation becomes detectable via [PublicKeyCredential.getClientCapabilities\(\)](#).

JavaScript

```
async function checkImmediateMediationAvailability() {  
  try {
```

```

const capabilities = await PublicKeyCredential.getClientCapabilities();

if (capabilities.immediateGet) {
  log("Immediate Mediation supported.");
} else {
  log("Immediate Mediation unsupported.");
}
} catch (error) {
  if (error instanceof TypeError) {
    log(
      "PublicKeyCredential.getClientCapabilities() is not supported on  
this browser.",
    );
  }
}
}

```

Testing

A. **(Optional)** Try the reference implementation on:

<https://deephand.github.io/webauthn-immediate-demo/>.

- a. Set both toggles on and click “sign in”.
- b. If you have passkeys and/or passwords saved, they will be displayed to you.
 - i. Choosing a saved password or passkey should log you in.
- c. If there are no passkeys or passwords saved, the site navigates to a sign-in page. You can “sign in” and it will allow you to create a passkey for subsequent use.
- d. **Note1:** This site doesn’t have an account backend so all sign-ins will be successful.
- e. **Note2:** Source code is available at <https://glitch.com/edit/#!/webauthn-immediate-demo>

B. **(Required)** Implement a prototype on your site (pre-production is OK!)

- a. Call `navigator.credentials.get`
- b. Set `mediation: 'immediate'`
- c. If you also want to request passwords, set `passwords:true`.
- d. Here is an example call

JavaScript

```
const cred = await
navigator.credentials.get({
  password: true,
  publicKey: {
    challenge,
    rpId: 'example.com'
  },
  mediation: 'immediate',
});
```

- e. **Note1:** We recommend you make this call after a button click e.g. the “sign in” button on your site.
- f. **Note2:** If there are no immediately available passkeys or passwords, the API method rejects the returned promise with `NotAllowed`. The intention in this case is that the test site navigates to the existing sign-in page. Users without credentials available at this point should see no change in the sign-in flow.
- g. **Note3:** We recommend including `password: true`, because most users are likely to have a saved password today. They can start benefitting from this experience immediately. If you couple this experience with [conditional create](#), you’ll be able to upgrade your users to passkeys. The next time they log in (and you call this API), we’ll prefer the passkey over the password
- h. **Note4:** This mode does not work with `allowCredential` lists.

WebAuthn immediate mediation demo

Important Information:

- Try this in the latest **Chrome Canary**.
- Enable **chrome://flags/#enable-experimental-web-platform-features** for the feature.



Immediate Mediation



Request Password

Sign In

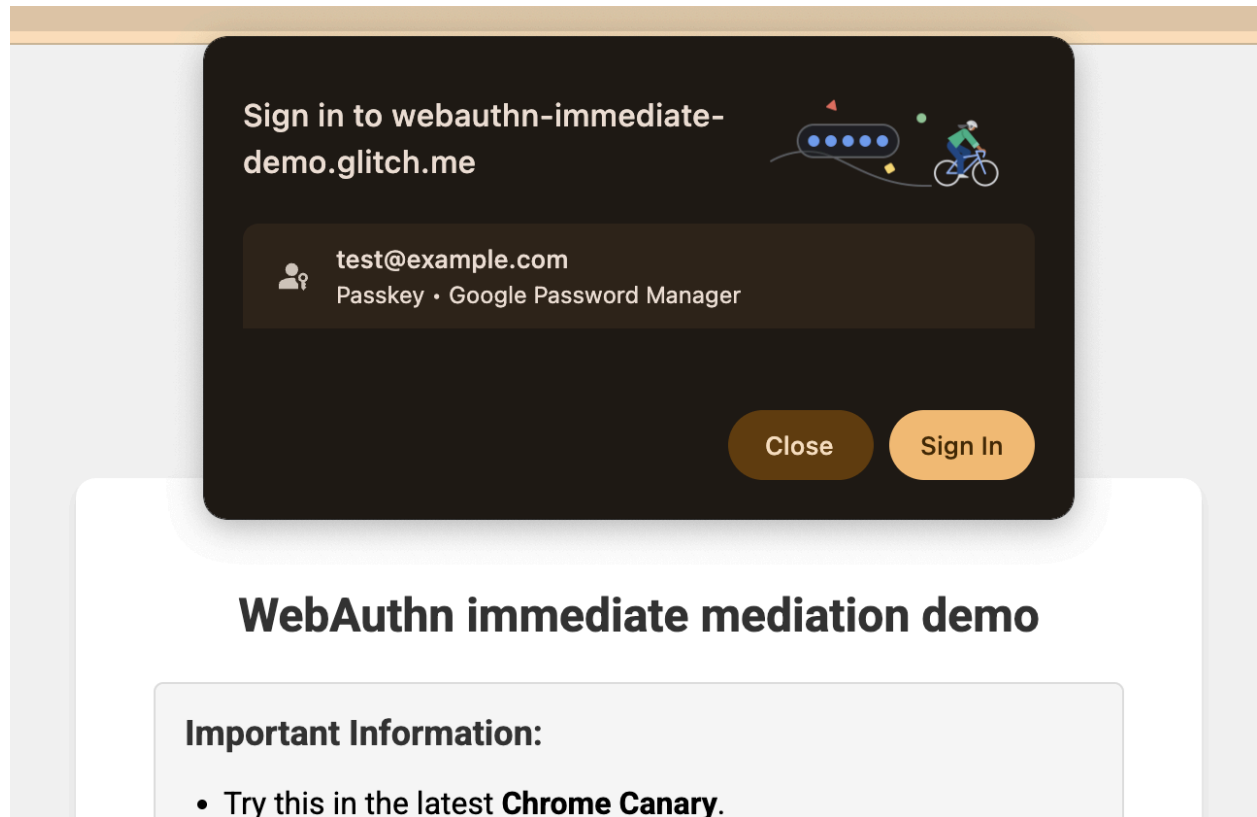
Instructed by derinel@google.com, implemented by Gemini

Flow 1: Sign in without password or passkey

On the test site, if the user has no passkeys or passwords available for this site already, clicking the "Sign In" button will take the user to a form that requests a username and then an OTP (anything typed into these fields will result in a "successful" sign-in).

Flow 2: Sign in with an immediately-available local passkey

After "signing in", you can create a passkey on the site with a "Create Passkey" button. Returning to the home page and clicking "Sign In" again should trigger the Immediate Mediation UI offering the recently-created passkey:



Flow 3: Sign in with a local passkey or password

Add a username/password pair for the site manually into the password manager, and then return to the test site. Toggle the "Request Password" option and click the "Sign In" button. The Immediate Mediation UI should offer both the passkey and password as sign-in options:

Sign in to webauthn-immediate-
demo.glitch.me



test@example.com

Passkey • Google Password Manager



OldSkoolPW

Password • Google Password Manager



Close

Sign In

WebAuthn immediate mediation demo