

[Image: Sign-In Doors on a Static Page]

Sign-In Doors on a Static Page

TileDown's newest tile renders a members sign-in on a fully static page. One directive block becomes Sign in with Apple, Sign in with Google, and an email magic-link form, all pointed at a site-owned auth API:

```
:::tile signin
endpoint: https://example.com/archive
providers: apple, google
title: I am a subscriber
:::
```

endpoint is required and can be an absolute URL or a site-absolute path for same-domain routing. providers is optional and ordered; the default is both, and an unknown value fails the build with a named error instead of a silent skip. The email form is always present and always last, as the universal fallback.

The invariant

The tile authenticates nobody and stores nothing. Each provider button is a plain link to the backend's OAuth route at {endpoint}/oauth/{provider}, with brand-guideline markup as engine-owned HTML and CSS. The email form POSTs {email, website} to {endpoint}/signin and renders the check-your-inbox acknowledgment, identical for known and unknown addresses. There are no tokens, no cookies, no localStorage writes, and no provider SDKs in the page; the engine's browser gate asserts all four.

Live demo

The tile below is real engine output. The demo endpoint has no backend behind it, so the provider doors land on a coming-soon page and a submitted email reports failure while keeping your input. That is the degradation contract: the page never breaks, and the tile never pretends.

```
:::tile signin endpoint: /demo/auth title: Members sign in (demo) :::
```

What the backend owns

Everything interesting: the OAuth dance, session minting, identity linking, and the membership check against the subscriber table. The tile only needs GET {endpoint}/oauth/{provider} to start a provider flow, POST {endpoint}/signin to accept the magic-link request, and CORS that allows the publishing site's origin. Swap the backend and the page does not change.

The [Feature Tour](#) now includes the tile alongside its siblings.