

Learn CIAM by example

with 4 recipes to improve your
app security and user experience



Auth0 by Okta

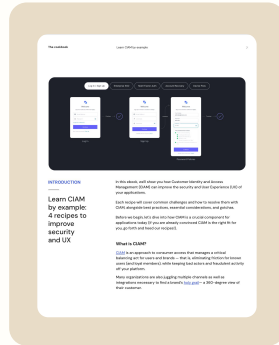
These materials and any recommendations within are not legal, privacy, security, compliance, or business advice. These materials are intended for general informational purposes only and may not reflect the most current security, privacy, and legal developments nor all relevant issues. You are responsible for obtaining legal, security, privacy, compliance, or business advice from your own lawyer or other professional advisor and should not rely on the recommendations herein. Okta is not liable to you for any loss or damages that may result from your implementation of any recommendations in these materials. Okta makes no representations, warranties, or other assurances regarding the content of these materials. Information regarding Okta's contractual assurances to its customers can be found at okta.com/agreements.

Recipes

Introduction

07

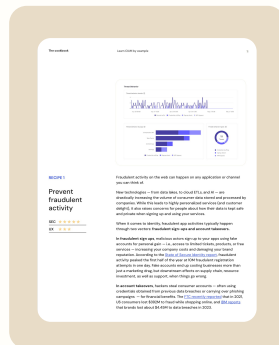
Learn CIAM by example: 4 recipes to improve security and UX



Recipe 1

11

Prevent fraudulent activity



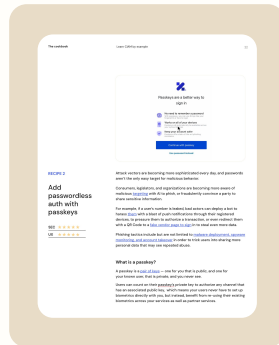
SEC ★★★★★

UX ★★☆☆☆

Recipe 2

22

Add passwordless auth with passkeys



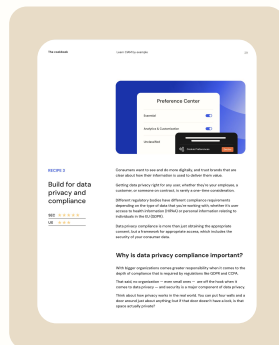
SEC ★★★★★

UX ★★★★★

Recipe 3

29

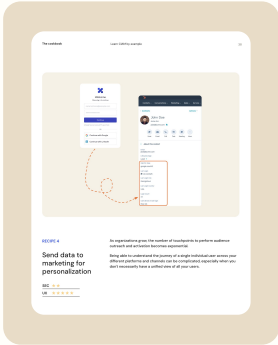
Build for data privacy and compliance



SEC ★★★★★

UX ★★☆☆☆

Recipes



Recipe 4

38

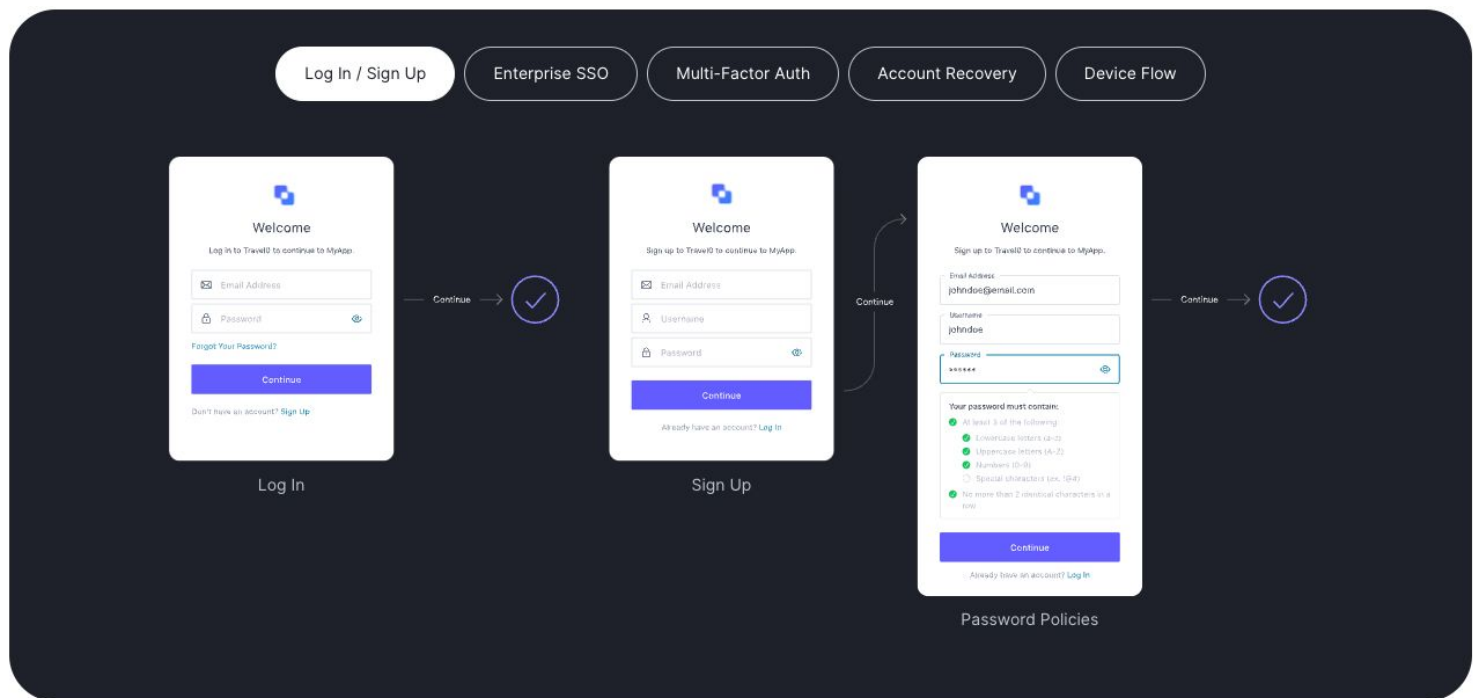
Send data to marketing
for personalization

SEC

★ ★

UX

★ ★ ★ ★ ★



INTRODUCTION

Learn CIAM by example: 4 recipes to improve security and UX

In this ebook, we'll show you how Customer Identity and Access Management (CIAM) can improve the security and User Experience (UX) of your applications.

Each recipe will cover common challenges and how to resolve them with CIAM, alongside best practices, essential considerations, and gotchas.

Before we begin, let's dive into how CIAM is a crucial component for applications today (if you are already convinced CIAM is the right fit for you, go forth and heed our recipes!).

What is CIAM?

[CIAM](#) is an approach to consumer access that manages a critical balancing act for users and brands — that is, eliminating friction for known users (and loyal members), while keeping bad actors and fraudulent activity off your platform.

Many organizations are also juggling multiple channels as well as integrations necessary to find a brand's [holy grail](#)— a 360-degree view of their customer.

Why is this important? With a centralized user profile, organizations can promote better security, because they know who is actually accessing their application, eliminate silos, and close in on platform gaps that leave brands vulnerable to fraud.

CIAM helps you define a central user profile from which you can securely tack on any number of integrations that you may use in your business everyday.

There are so many CIAM solutions on the market that cover a host of security and marketing use cases that could be common or necessary for your own business.

Do I really need CIAM (aka *more software*)?

You built an app. And your app went viral! That's awesome!

Once you go viral, the holistic customer profile challenge can become all the more steep, because, when user activity volume increases, it can be hard to tell who's a real customer and who is potentially a bot deployed for the purposes of defrauding you and your customers.

[Your customers are now also expecting more fun and secure ways to engage with your brand](#), and CIAM serves to support UX with clear and friendly communication from your brand when it benefits users to sign in or sign up to access your products and services.

And with that, your application gets two new and conflicting missions: reduce app friction to bring even more customers in (aka improve UX) while bolstering security to keep your consumers safe and private, prevent breaches, and even bring B2B contracts.

Organizations can leverage CIAM to learn more about their own or another brand's (like a partner's) customer profile, and promote good data hygiene with **deduplication and unification of the customer identity for your teams**.

CIAM is also shown to [decrease friction at checkout](#), and, in the case of Okta CIC, [cut down on bad bot activity by ~79%](#).

CIAM solutions range from basic to advanced, but, at the core, CIAM is the front door of your brand, with different ways of addressing authentication, alongside varying degrees of fraud prevention software.

Brands that are expanding their businesses get the following benefits from CIAM:

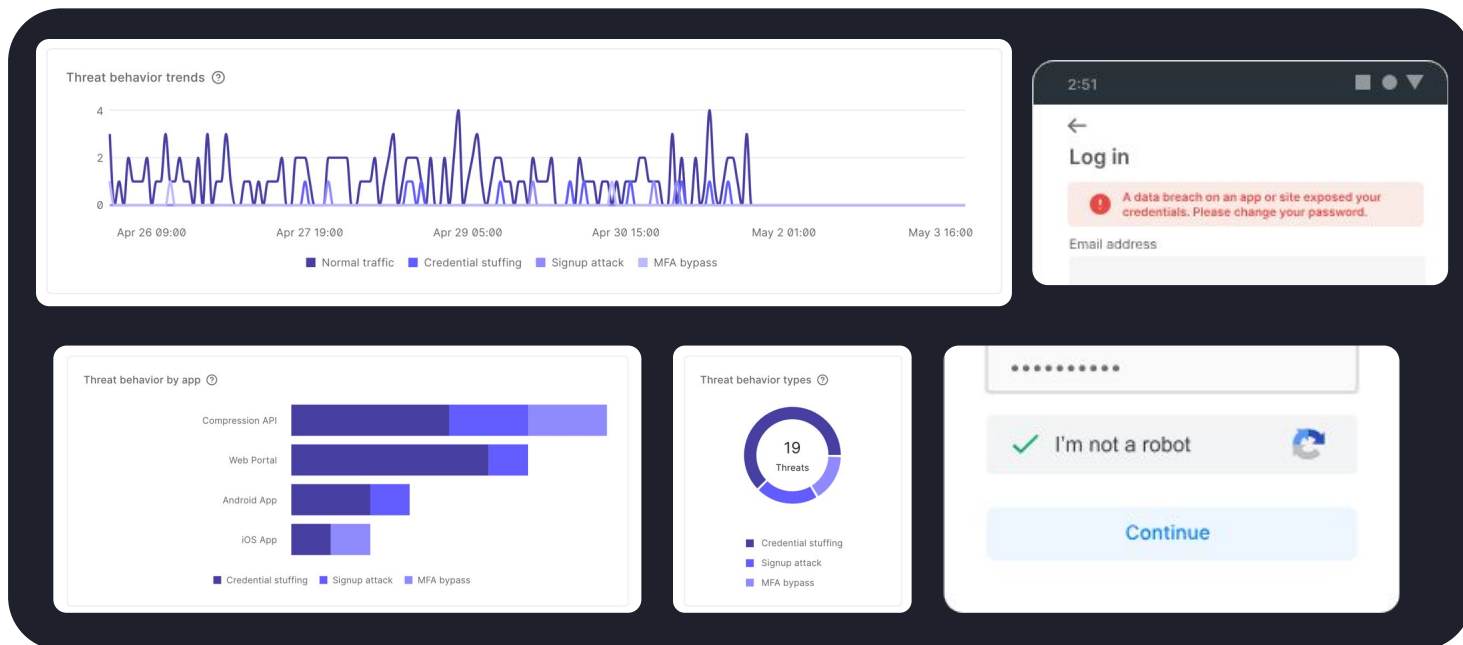
1. **Seamless integration** with your second- or third- party services,
2. **Manage the risks** of incorporating those services into your channels,
3. **Address more complexity** in the user journey.

With CIAM, you can convert more legitimate users, faster, because you made it easier for them to sign up, login, and self-service common account operations — your customers are safeguarded from fraudulent activity, and teams can maintain a sanitized customer profile.

How does this CIAM thing work, then?

That's where our recipes come in! Each recipe explores a real scenario, and how CIAM can help.

In the next chapter, we will explore our first recipe: [Prevent fraudulent sign ups and account takeovers.](#)



RECIPE 1

Prevent fraudulent activity

SEC ★★★★★

UX ★★☆☆☆

Fraudulent activity on the web can happen on any application or channel you can think of.

New technologies — from data lakes, to cloud ETLs, and AI — are drastically increasing the volume of consumer data stored and processed by companies. While this leads to highly personalized services (and customer delight), it also raises concerns for people about how their data is kept safe and private when signing up and using your services.

When it comes to identity, fraudulent app activities typically happen through two vectors: **fraudulent sign-ups** and **account takeovers**.

In fraudulent sign ups, malicious actors sign up to your apps using fake accounts for personal gain — i.e., access to limited tickets, products, or free services — increasing your company costs and damaging your brand reputation. According to the [State of Secure Identity report](#), fraudulent activity peaked the first half of the year at 10M fraudulent registration attempts in one day. Fake accounts end up costing businesses more than just a marketing drag, but downstream effects on supply chain, resource investment, as well as support, when things go wrong.

In account takeovers, hackers steal consumer accounts — often using credentials obtained from previous data breaches or carrying over phishing campaigns — for financial benefits. The [FTC recently reported](#) that in 2021, US consumers lost \$392M to fraud while shopping online, and [IBM reports](#) that brands lost about \$4.45M to data breaches in 2023.

Even if you don't have your own security operations center (SOC) to keep your systems safe, you can still make smart investments in technology to mitigate fraud, and carry your brand into the future, and even the metaverse (yes, people will still want to order pizza in the metaverse).

Why fraudulent identity activities are so hard to track

Fraudulent identity activities are hard to track because attacks are becoming cheaper and more sophisticated just as the technology that's trending today.

Cybercriminals are rapidly adopting low-code/no-code service to launch profitable attacks faster and regardless of their technical skills. This trend not only increases the frequency of attacks but also amplifies their impact on businesses year over year. [Since 2020, it is reported that organizations take at least 200 days to recover from a security breach.](#)

Legislation is still working on defining the parameters of what digital safety means for consumers and brands, not only on your apps and websites, but for the entire web.

Phase 1

Know who your users are and sign them in

With the rise in malicious identity threats [like phishing or social engineering](#), consumers are expecting brands to balance business outcomes and data privacy when it comes to the customer experience (CX), and technology and legislation are changing to support consumer safety on the web.

To accommodate the ongoing flux, brands can communicate through their platform's UX and user interface (UI), and let users know when it benefits them to sign up, and how your brand delivers value through their self-identification and your cybersecurity practices.

This post will cover basic Auth0 by Okta features with a full working demo, so you can bootstrap balancing security with UX.

If you haven't done so already, [sign up](#) for a free account, and set up your tenant to cover the use cases that make most sense for you and your organization.

Recipe

Ingredients:

- [An Auth0 Tenant](#)
- A sample application integrated to Auth0 (you can [use one of our quickstarts](#))

So, you've never secured an App using customer identity (CIAM)? We got you.

Apps and APIs can use CIAM for identity tasks — like sign-ups, social log in, authentication, authorization, and logouts. It's all delivered through open-source SDKs, APIs, and protocols, just like other services you might use (think Datadog for observability, Twilio for SMS, or Stripe for payments).

Here's how a login works with CIAM:

1. User tries to access a function requiring a session.
2. Your app uses the Auth SDK to check for a valid session. If none exists, the user is redirected to the CIAM system for authentication.
3. The user logs in.
4. The CIAM system handles authentication, authorization, and auditing, then redirects the user back to your app.
5. Your app uses the Auth SDK to access user details from the session and enable the requested functionality.

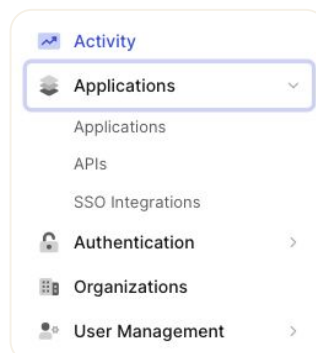
[CIAM gives you the ability to tweak the login process](#) — i.e. UX, social login, biometric auth, bot protection — to ensure only the right people have access to your app without implementing user friction.

Add CIAM to your application

This may be the most intimidating part of the entire project (starting is hard!), even if you have a little bit of coding know-how —so we are here to walk you through the actual steps you need to take in order to deploy your new application with Auth0.

1. **In the Auth0 Dashboard, you can create a login or sign up for almost any application, machine, or API you can think of.**

*Navigate to **Applications > Applications***



Select

+ Create Application

There are tons of pre-built templates that can help you get started, and customize for your own use cases — but, for the purposes of this tutorial you can create a new application instance for a single page application (SPA).


Create application

Name *

My App

You can change the application name later in the application settings.


Choose an application type



Native

Mobile, desktop, CLI and smart device apps running natively.


e.g.: iOS, Electron, Apple TV apps



Single Page Web Applications

A JavaScript front-end app that uses an API.


e.g.: Angular, React, Vue



Regular Web Applications

Traditional web app using redirects.

e.g.: Node.js Express, ASP.NET, Java, PHP



Machine to Machine Applications


CLIs, daemons or services running on your backend.

e.g.: Shell script

Cancel

Create

2. In the Auth0 Dashboard, you can create a login or sign up for almost any application, machine, or API you can think of.

Environment variables (.env.local)	Auth0 Dashboard
<code>AUTH0_SECRET='LONG_RANDOM_VALUE'</code>	<p>This should be a generated string, but, for testing, you can put whatever you'd like.</p> <p>For production, you can create a generated string here: https://jwt.io</p>
<code>AUTH0_ISSUER_BASE_URL='https://<your-domain-here>.auth0.com'</code>	<p>The url of your Auth0 tenant domain, your Auth0 application's Client ID, and Client Secret can all be found in one place.</p>
<code>AUTH0_CLIENT_ID='aP3UzAmNXTTfJUeGLISRbTZC1UUZE3kT'</code>	<p>Navigate to Applications > Applications > Your Application, and capture the following information:</p>
<code>AUTH0_CLIENT_SECRET='LongAPPsecret'</code>	 <p>The Client Secret is not base64 encoded.</p>
<code>AUTH0_BASE_URL='http://localhost:3000'</code>	<p>Once you've downloaded the project, you will notice that the deployment is locally hosted, i.e. <code>http://localhost:3000</code>.</p> <p>As you test, you may have your own server information that you may want to reference.</p>

3. So far, we have registered your app.

To complete your configuration, you need to inform Auth0 on where to redirect a user after a successful login and log out (callback URL).

This step has tripped up many a developer (copy paste gets the best of us), but it's an easy step if you know exactly what to type for the callback. In this case, it is

`http://localhost:3000/api/auth/callback`.

Allowed Callback URLs

http://localhost:3000/api/auth/callback

After the user authenticates we will only call back to any of these URLs. You can specify multiple valid URLs by comma-separating them (typically to handle different environments like QA or testing). Make sure to specify the protocol (`https://`) otherwise the callback may fail in some cases. With the exception of custom URI schemes for native clients, all callbacks should use protocol `https://`. You can use [Organization URL](#) parameters in these URLs.

Allowed Logout URLs

http://localhost:3000/api/auth/callback

A set of URLs that are valid to redirect to after logout from Auth0. After a user logs out from Auth0 you can redirect them with the `returnTo` query parameter. The URL that you use in `returnTo` must be listed here. You can specify multiple valid URLs by comma-separating them. You can use the star symbol as a wildcard for subdomains (`*.google.com`). Query strings and hash information are not taken into account when validating these URLs. Read more about this at <https://auth0.com/docs/authenticate/login/logout>

4. And now, the fun part

You can run `npm run dev` in your favorite IDE, and your application is deployed, with the login works.

Now that your application is integrated to customer identity, we can start turning security knobs and switches on to prevent fraudulent activity!

Phase 2

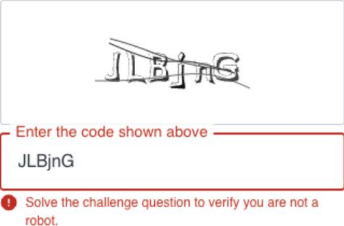
Prevent fraudulent sign-ups from bots

Marketers are on the [front lines of marketing fraud](#).

Immersive experiences that consumers get excited about will only be possible if you [use your CIAM solution](#) to help enforce security and keep bots out.

However, adding security can introduce friction that prevents your customers from using your product!

Here's an example of the friction:



Standalone Captchas slow bots down but frustrate customers :(

Having control over the timing of these safeguards is paramount to UX.

When your application uses Auth by Okta, you can easily control when consumers see a CAPTCHA, and implementation takes only 2 steps: 1.) Turn the bot protection on, and 2.) Select the risk level that will serve as threshold for the bot prevention:

1: Turn bot prevention on

Choose when to require CAPTCHA for flows with passwords

Never
Users are not required to complete a CAPTCHA to log in.

When Risky
Users are required to complete a CAPTCHA if the login is high risk.

Always
Users are always required to complete a CAPTCHA to log in.

2: Select the risk level

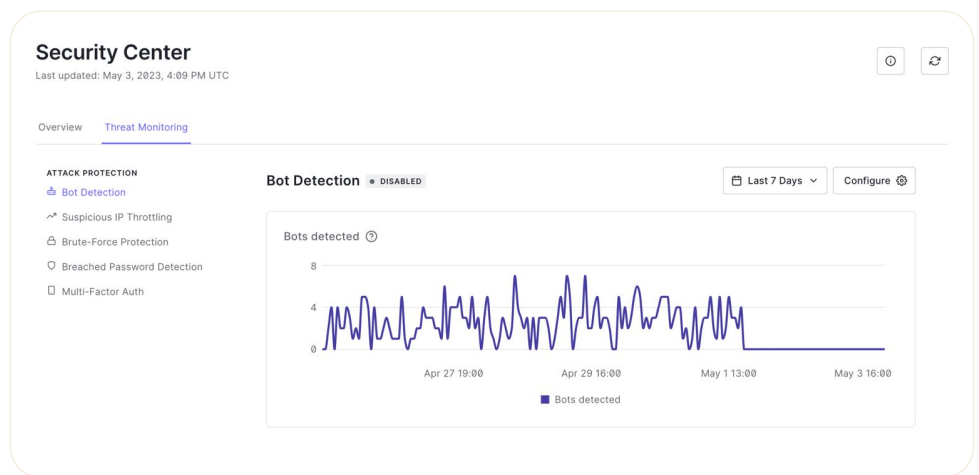
LowMedium (Default)High

BOT DETECTION LEVEL

Medium Security results in a moderate classification of bots and users seeing CAPTCHA. This provides a balance of security and user experience for your users.

The risk level works as the secret sauce for bot prevention, giving you the ability **to strike the perfect balance between preventing fraudulent sign up/take overs while not introducing friction to legit users.**

And, if you're not sure of what risk level to select, keep an eye at the [Security Center](#). It provides advanced real-time monitoring, available to enterprise customers, to help you see how the bot prevention is performing in your app:



Security Center: Bot detection monitoring

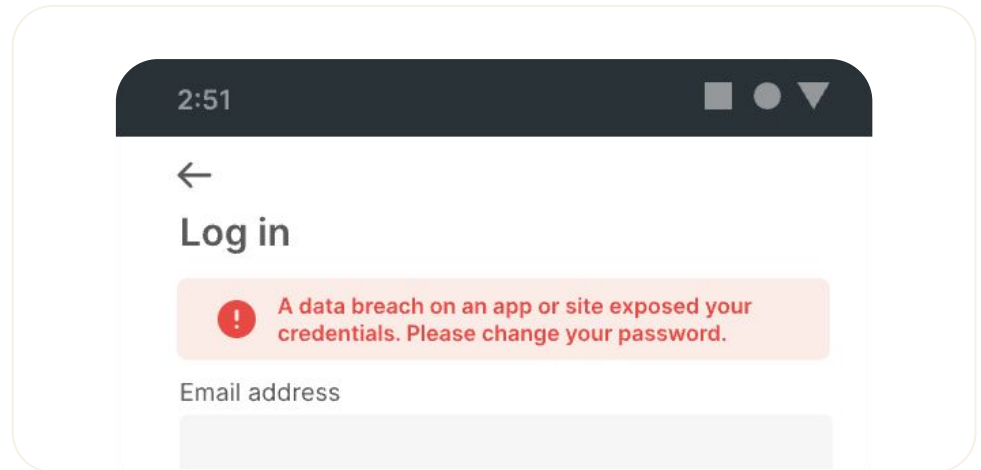
Phase 3

Prevent account takeovers from breached credentials

Consumers trust brands that take the time to ensure that they are notified about new or unfamiliar activity on their accounts.

To differentiate from phishing that mimics generic messaging, consumers are relying on organizations to tailor this messaging towards them, and not their junk folder.

[Auth0 by Okta offers one-click Integrations](#) in order to notify users and your admins in the event that anomalous activity occurs on your platform when it comes to their accounts and credentials.



These on-off switch features not only save time and investment developing the templates and infrastructure necessary to alert users to suspicious activity.

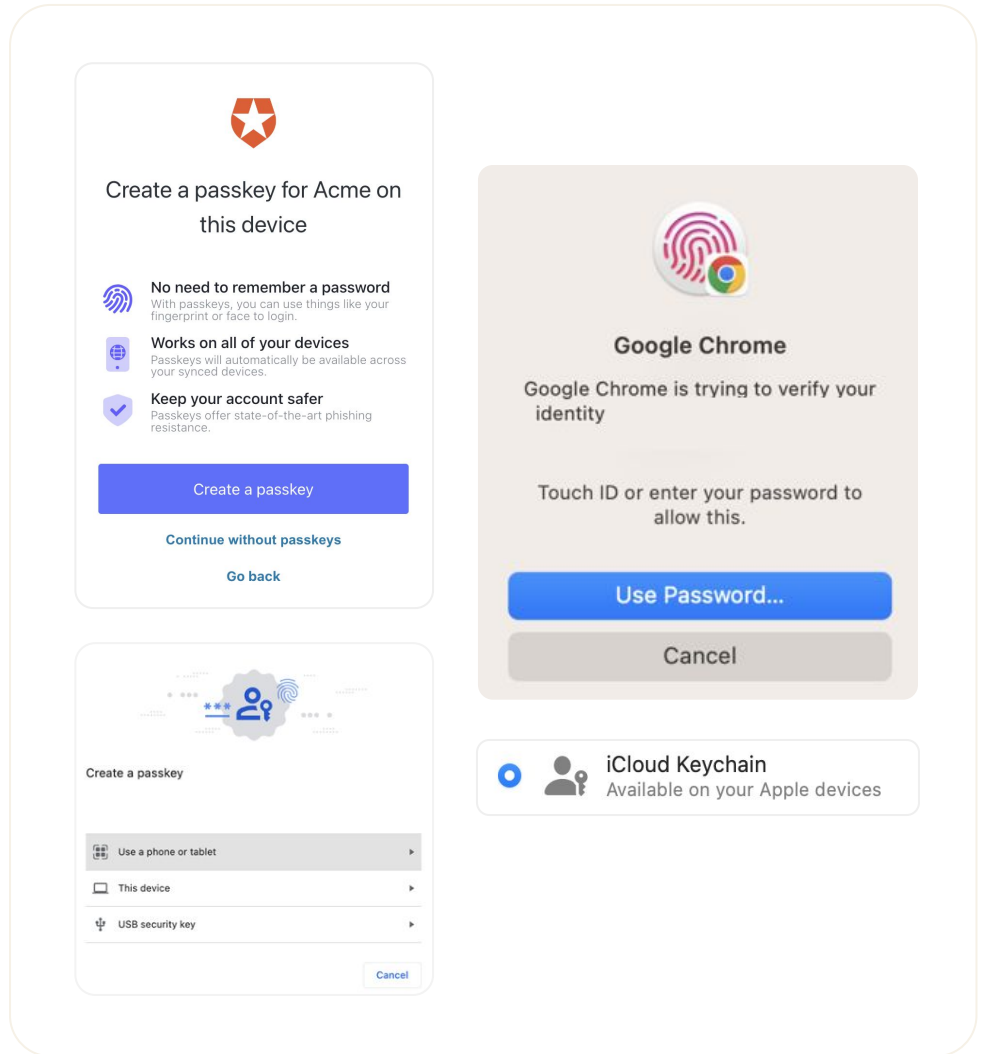
Also, teams can focus on making notifications personal to the user, so they are more likely to engage when something doesn't seem right.

With a variety of attack vectors being combined through cheap and easy means by bad actors on the web, Auth0 by Okta addresses them all out-of-the-box, with low technical investment and fast time-to-market.

Phase 4

Migrate away from passwords

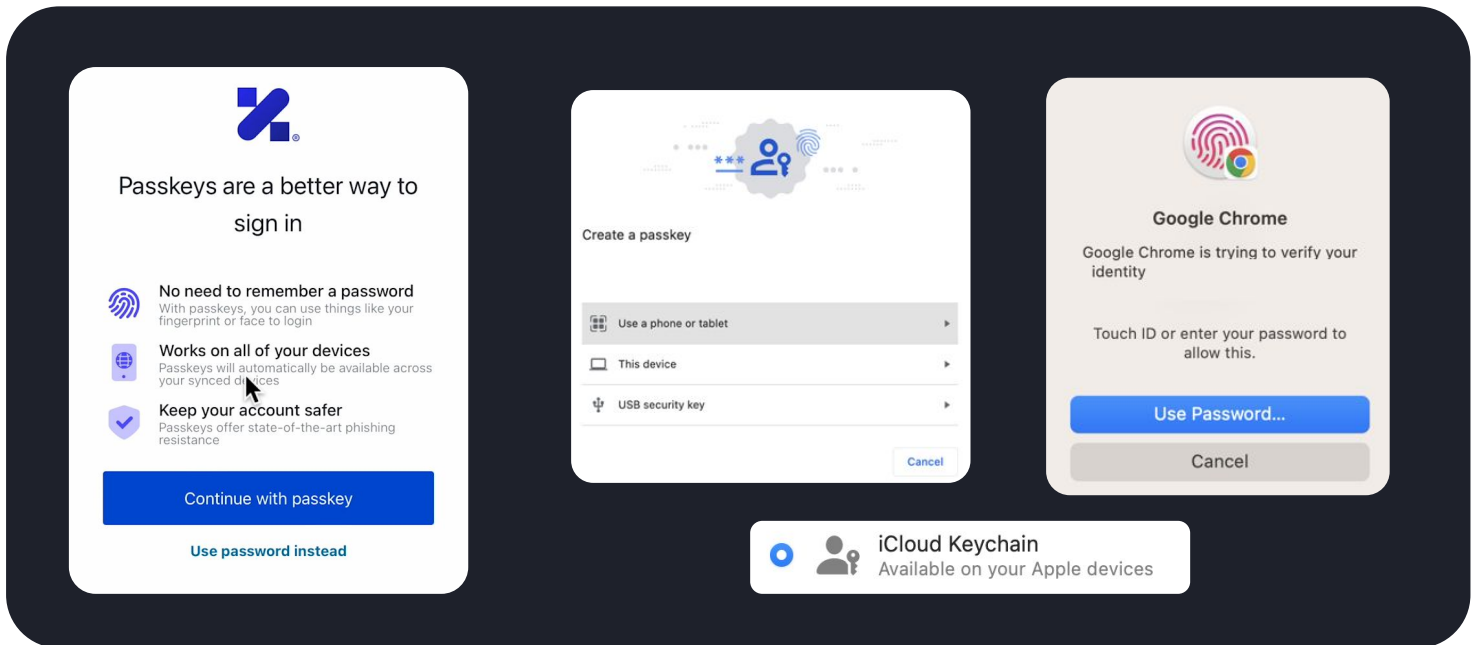
There is a new W3C and FIDO specification on the block, written to minimize login friction, and create phishing-resistant platforms — the standard is for a new type of factor, known as a [passkey](#), which is already in use by many service providers including Apple and Google.



Instead of your users making a facsimile of their biometrics every time they want to onboard, they can authorize a login using their existing face or fingerprint biometric on their device, in any transaction, on any platform they choose.

New Universal Login supports the use of passkey, which means users can skip passwords altogether.

In the next recipe, we will dive into passkeys with the recipe: [Add passwordless auth with passkeys.](#)



RECIPE 2

Add passwordless auth with passkeys

SEC ★★★★★

UX ★★★★★

Attack vectors are becoming more sophisticated every day, and passwords aren't the only easy target for malicious behavior.

Consumers, legislators, and organizations are becoming more aware of malicious [targeting](#) with AI to phish, or fraudulently convince a party to share sensitive information.

For example, if a user's number is leaked, bad actors can deploy a bot to harass [them](#) with a blast of push notifications through their registered devices, to pressure them to authorize a transaction, or even redirect them with a QR Code to a [fake vendor page to sign in](#) to steal even more data.

Phishing tactics include but are not limited to [malware deployment, spyware monitoring, and account takeover](#) in order to trick users into sharing more personal data that may see repeated abuse.

What is a passkey?

A passkey is a [pair of keys](#) — one for you that is public, and one for your known user, that is private, and you never see.

Users can count on their passkey's private key to authorize any channel that has an associated public key, which means your users never have to set up biometrics directly with you, but instead, benefit from re-using their existing biometrics across your services as well as partner services.

Developers can rest easy knowing that bad actors can't do anything with their hosted public key, because there is no consumer profile information or credentials associated with a public key — only a user-authorized private key can link up with the public key to grant access.

With [passkeys](#), organizations can sign in consumers across networks using their smartphone's existing mobile technology, making it possible to authenticate almost anywhere using the same biometric or pin that they use to unlock their devices.

Why are passkeys so hot right now?

Customers want to spend less time logging in.

The way we send texts, make calls, and even post social media updates is the same technology that service providers like [Apple](#) and [Google](#) have baked into their products to make it easier and faster to login to their ecosystems.

Now, organizations can make their own [vendor ecosystems](#), and offer users more secure ways to sign in and manage their personal data without using a password. Passkeys can also [speed up authentication by 2.6x](#) on your platform.

Passkeys deliver better UX that gives organizations more control over their digital properties across systems and devices, with the added benefit for users of avoiding passwords altogether.

Fundamentally, passkeys make consumers happy because they can change devices in different contexts and environments and maintain a seamless experience, and organizations can boost their funnel.

While it may seem intimidating to add yet another digital credential to the list, passkeys are backed by strong cryptographic standards that are worth the hype, and any organization can add passkeys to their anti-phishing strategy with Auth0 by Okta.

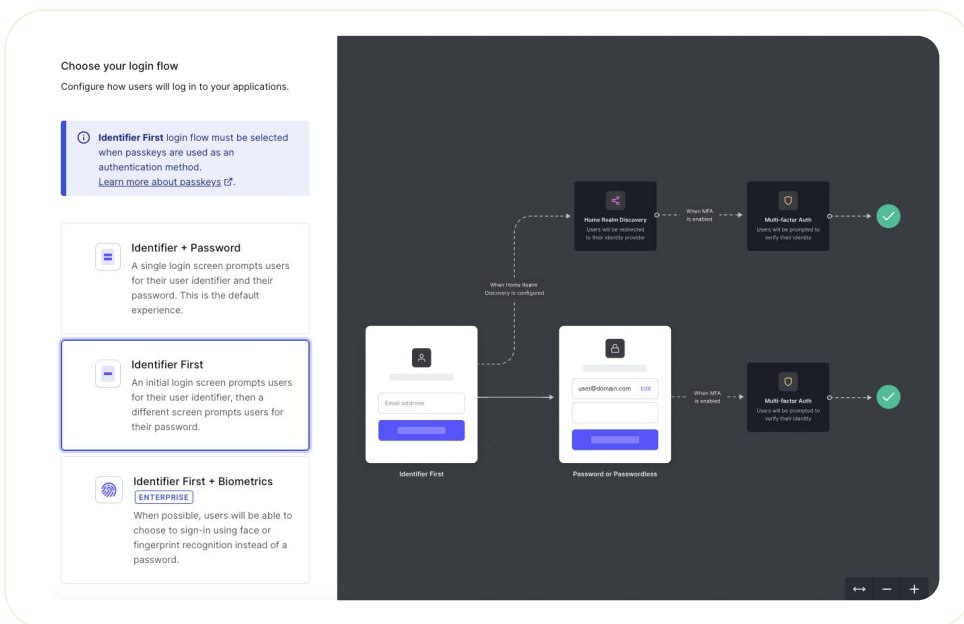
Recipe

Ingredients

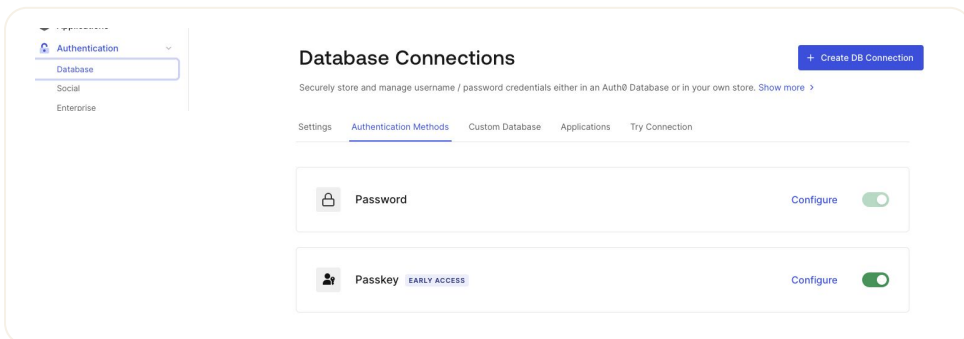
- [An Auth0 Tenant](#)
- [A sample application to test sign up, login, and logout](#)

Passkeys are super easy to set up. Just a few steps in the Auth0 dashboard, and your cloud-hosted New Universal Login will update the rest, with passkeys available for all users.

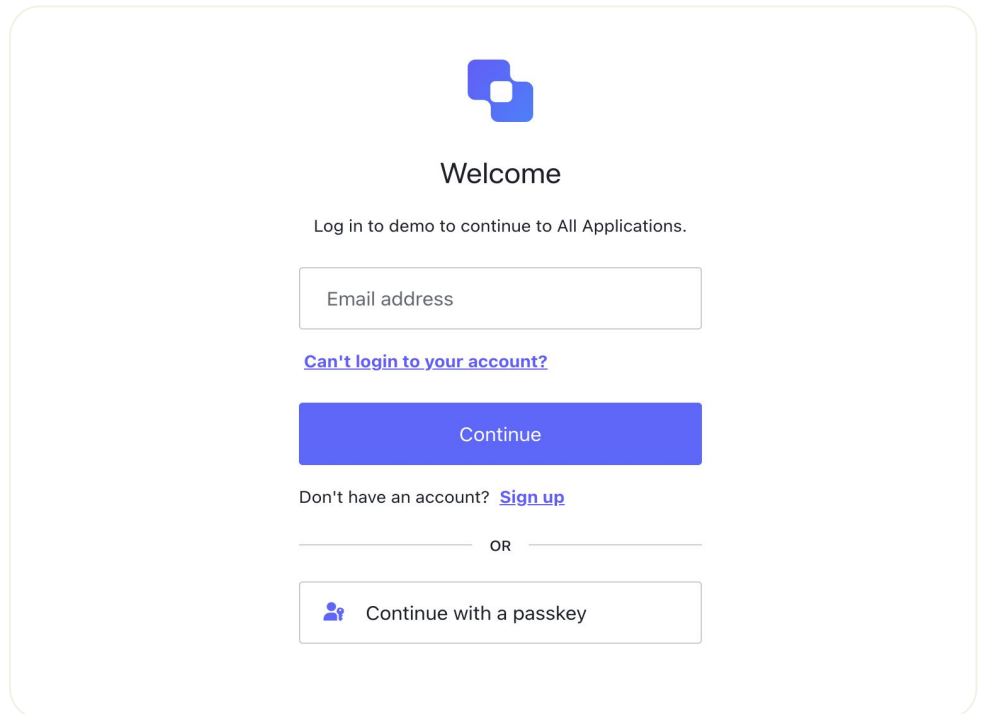
1. Navigate to **Authentication > Authentication Profile**, and select **Identifier First**.



2. Navigate to **Authentication > Database** and select **Create DB Connection**. Name it "Passkeys" and click **Create**. Select the **Authentication Methods** tab on the next screen and enable **Passkey**.



3. And that's it! When you go to login or sign up, you will see passkey as an option, with all [UX, and secure authentication](#) best practices baked in:



Welcome

Log in to demo to continue to All Applications.


Email address

[Can't login to your account?](#)

Continue

Don't have an account? [Sign up](#)

OR

 Continue with a passkey

Passkeys: the customer friendly security

Passkeys make it easy for users to safely traverse multiple networks and channels, because passkeys cut through multiple vendor networks, without sharing credentials, including passwords.

Passkeys are never shared outside of their device or services, and capture a user's consent (double tapping that side button) to access their digital properties, without entering a password.

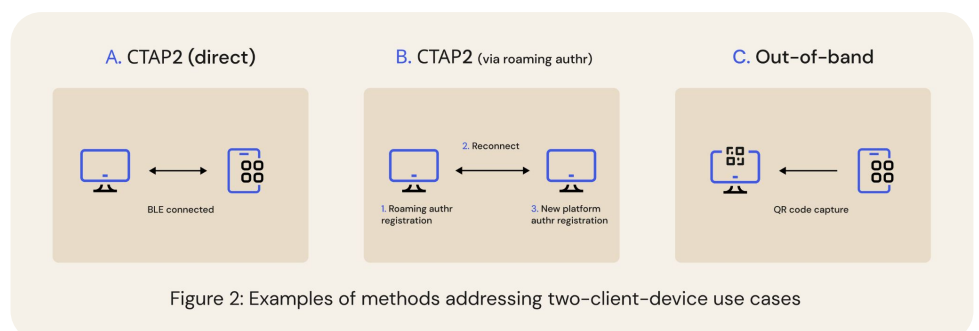


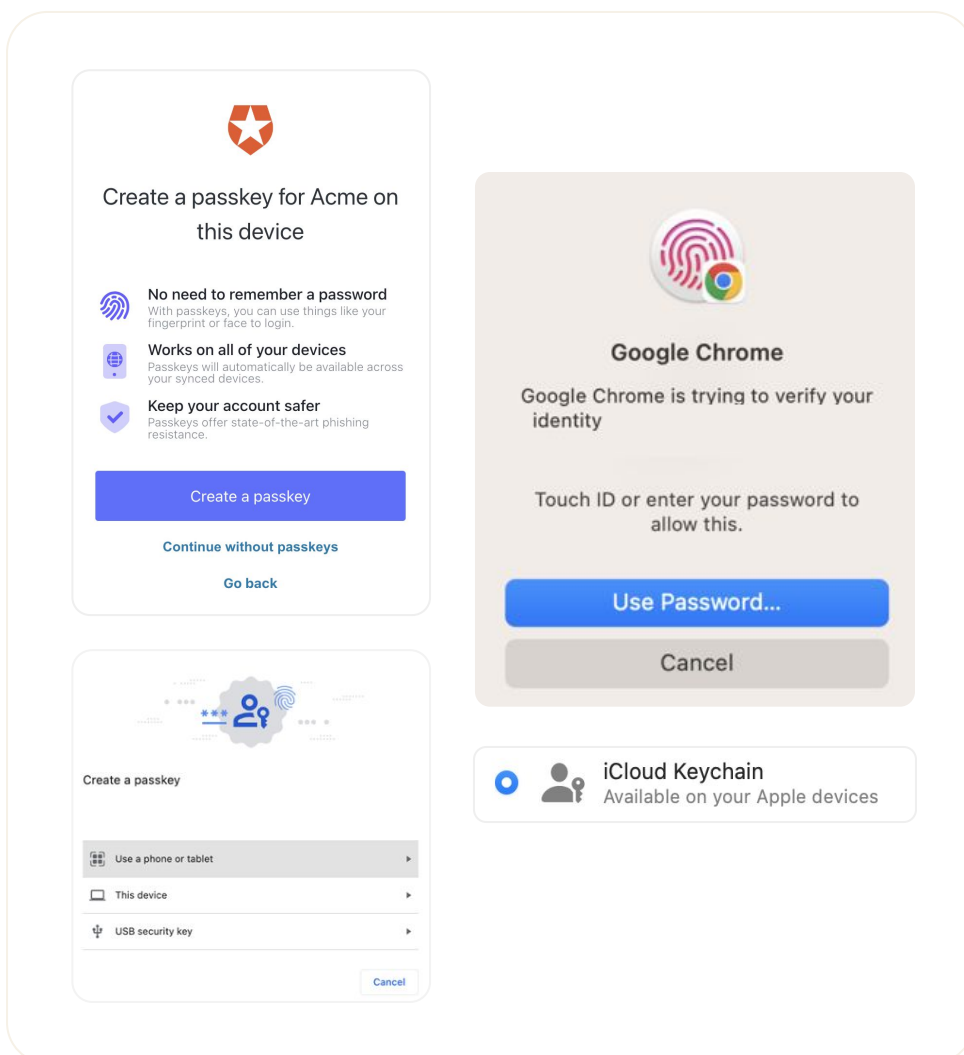
Figure 2: Examples of methods addressing two-client-device use cases

Software-bound

Passkeys can be used to authenticate without passwords across devices and services to continue streaming your favorite show from your smartphone to your tablet.

Let's say your user signed up for your streaming service using their Google credentials, but they use their Apple ID for everything else because they have an iPhone and an iPad.

A user's iPhone passkey can be used (with their consent) to create a passkey for that streaming service on their iPad for their Google account, without any of the steps that are involved with normal biometrics enrollment and sharing between vendors is replaced by a synced passkey.



Device-bound

Passkey hardware, like a Yubikey, is the gold standard for phishing-resistance and consumer-available security — in fact, when performed over near-field communication (NFC), it is the most secure way to use a passkey, because the passkey never leaves the device, and the use of a physical device accommodates proof of presence.

A device-bound passkey can be used to perform [cross-device authentication](#) in proximity to each other — and instantly sync, sign in, (and start streaming) across devices, and is considered the most secure passkey pattern.

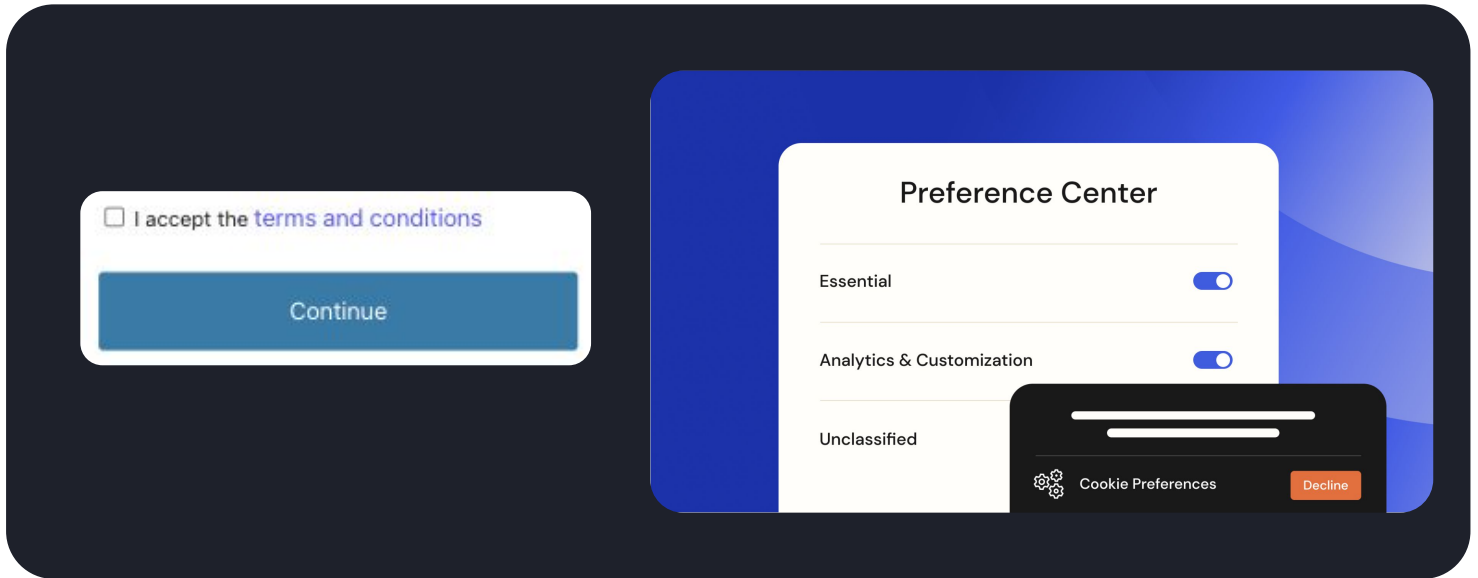
Brands can protect their perimeter by encouraging the use of passkey hardware, and, [like one media giant](#), see higher conversion rates and a dramatic decrease in account service requests, all while protecting against malicious activity, and delivering a secure, frictionless experience that consumers love.

To learn more about device-bound passkeys, [check out our post on how to set up with Auth0](#).

What's next?

Congratulations on adding passwordless to your apps with passkey! Now, we're ready to check in on data privacy! To fully protect your customers' information and comply with regulations like GDPR, HIPAA, and CCPA, your app needs to deliver additional features like user consent, data auditing, and a preference center.

In our upcoming recipe, [we'll delve into essential elements to get your app inline with data privacy](#).



RECIPE 3

Build for data privacy and compliance

SEC ★★★★★

UX ★★★

Consumers want to see and do more digitally, and trust brands that are clear about how their information is used to deliver them value.

Getting data privacy right for any user, whether they're your employee, a customer, or someone on contract, is rarely a one-time consideration.

Different regulatory bodies have different compliance requirements depending on the type of data that you're working with, whether it's user access to health information (HIPAA) or personal information relating to individuals in the EU (GDPR).

Data privacy compliance is more than just obtaining the appropriate consent, but a framework for appropriate access, which includes the security of your consumer data.

Why is data privacy compliance important?

With bigger organizations comes greater responsibility when it comes to the depth of compliance that is required by regulations like GDPR and CCPA.

That said, no organization — even small ones — are off the hook when it comes to data privacy — and security is a major component of data privacy.

Think about how privacy works in the real world. You can put four walls and a door around just about anything; but if that door doesn't have a lock, is that space actually private?

Phase 1

Implement an audit log

[Per GDPR](#) and [CCPA](#), organizations great and small are obliged to certain auditing practices, and the foundation of these practices begins with an audit compliance layer, or an audit log.

Unlike SIEM or system logging tools, audit logging is set up to relay the impact of a security incident in a readable prompt, like a historical record of events, to assess the risk associated with an individual's or group's actions on your platform, as compared to the permissions they have, and raise a red flag when they don't match.

With Auth0 by Okta, you don't need to build your own audit log stream — our logs are ready for the auditing out-of-the-box, with [several compliance](#) certifications built in, to help support [compliance readiness](#).

But, that's only the beginning.

Audit logs tell you what parts of your platform are frequented, and what data is visible to your lowest common denominator.

It's up to organizations to mobilize these insights into tangible protection for their consumers, and provide tools that give them control over their data.

Phase 2

Consent

Data privacy laws may require organizations to obtain appropriate consent before processing personal data. With Auth0 by Okta professional and enterprise, customers can obtain consent using Custom Prompts.

Custom Prompts is built on the [Liquid](#) template language, designed to give developers enhanced control over the login and signup experience, with partial templates at various [entry points](#).

Basically, with a bit of HTML, CSS, and Javascript, teams can bootstrap their consent efforts with Auth0 by Okta's cloud-hosted Universal Login.

These partial templates can not only accomplish granular consent, but capture other information at different points in the authentication journey, powered by Auth0 by Okta Actions.

In this recipe, we will add a [signup prompt](#) using Actions.

Recipe

Ingredients

- Universal Login
- [Custom Domain](#) (Branding > Custom Domains)
- Custom [Page Template](#)

1. Let's load our partial consent template into our login with an [API call](#). Here's the consent partial:

```
<div class="ulp-field">
  <input
    type="checkbox"
    name="ulp-terms-of-service"
    id="terms-of-service">
  <label for="terms-of-service">
    I accept the
    <a href="https://example.com/tos">terms and conditions</a>
  </label>
</div>
```

Add this to a curl command, and go:

```
# Add your own tenant info
URL='https://TENANT.ENVIRONMENT.auth0.com/api/v2/prompts/signup/partials'
TOKEN='eyJhbGciOi...'

curl -X PUT \
-H 'Content-Type: application/json' \
-H "Authorization: Bearer $TOKEN" \
-d '{"signup":{"form-content-end":"'<div class= 'ulp-field '>
<input type= 'checkbox ' name= 'ulp-terms-of-service ' id=
'terms-of-service '> <label for= 'terms-of-service '> I accept the <a
href= 'https://example.com/tos '>terms and conditions</a> </label>
</div> " \
"$URL"
```

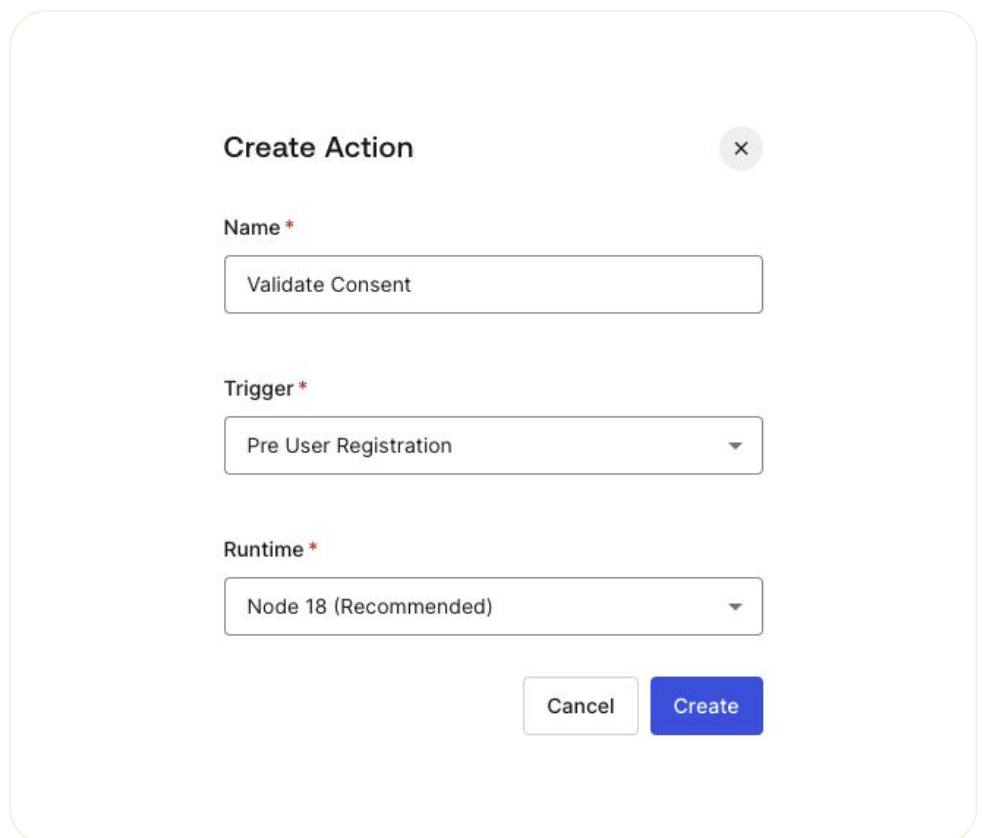
2. Since Universal Login is already ready to [support consistent, branded UX](#), your custom partials will fit right in with the rest of the UI:



☐ I accept the [terms and conditions](#)

Continue

3. In order to secure this client-side code, we need to take the extra step of creating server-side validation, in the form of a [pre user registration Action](#), by navigating to **Actions > Library > Build Custom**:



Create Action ×

Name ^{*}

Validate Consent

Trigger ^{*}

Pre User Registration ▼

Runtime ^{*}

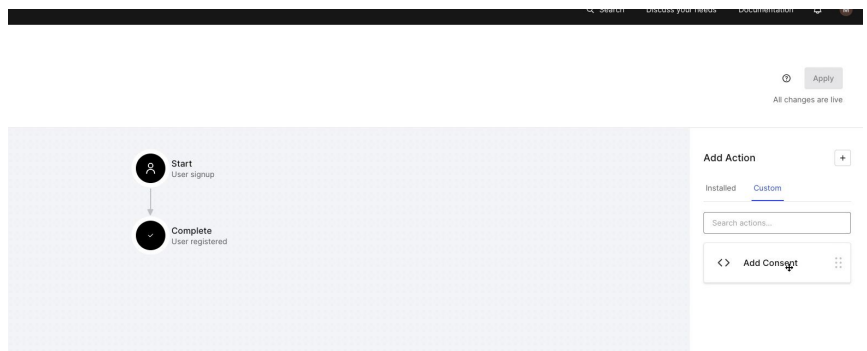
Node 18 (Recommended) ▼

Cancel Create

4. Add the following code to the Action, which will prevent the form from validation unless consent is given, then **Deploy** to save:

```
exports.onExecutePreUserRegistration = async
(event, api) => {
  const termsOfService =
event.request.body['ulp-terms-of-service'];
  if(!termsOfService) {
    api.validation.error("invalid_payload", "Please review the terms of
service.");
    return;
  }
  api.user.setUserMetadata("termsOfService", true);
};
```

5. Navigate to **Flows > Pre User Registration**, and add your **Custom Action**:



6. Now, when your user gives consent, this will be documented in their profile (**User Management > Users**), under `user_metadata`:

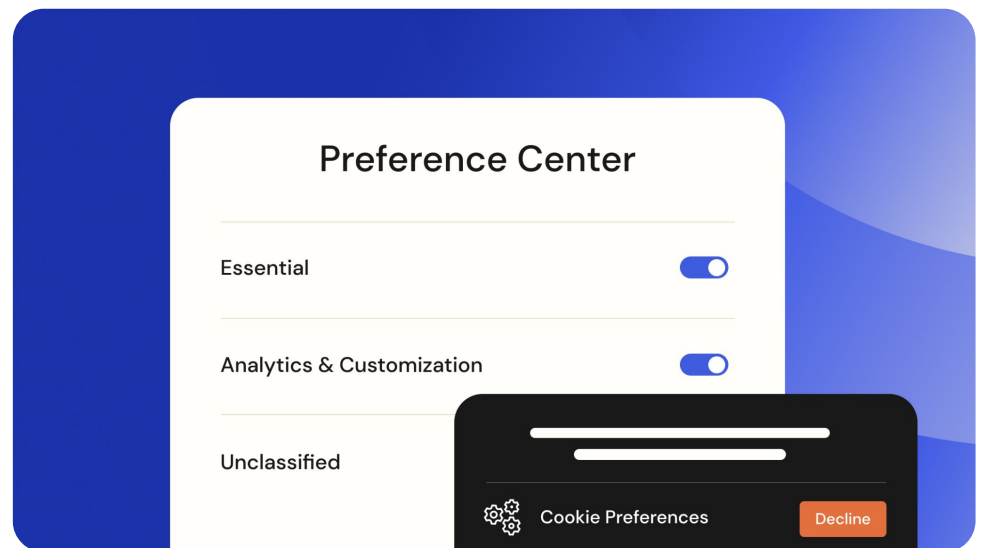
```
1 {
2   "termsOfService": true
3 }
```

Phase 3

Preference center

Data privacy compliance requires that users have not only clearly consented to the collection and use of their personal data, but be able to revoke it as well as access or correct their data and/or delete it from your platform.

A [Preference Center](#) is a place where users go to manage their, well, preferences, and this includes the newsletters they signed up for, and any other service in between that requires their explicit consent.



When organizations invest in an identity solution like Auth0 by Okta, Building preference centers are a breeze through the [Auth0 Management API](#).

Phase ∞

Data security

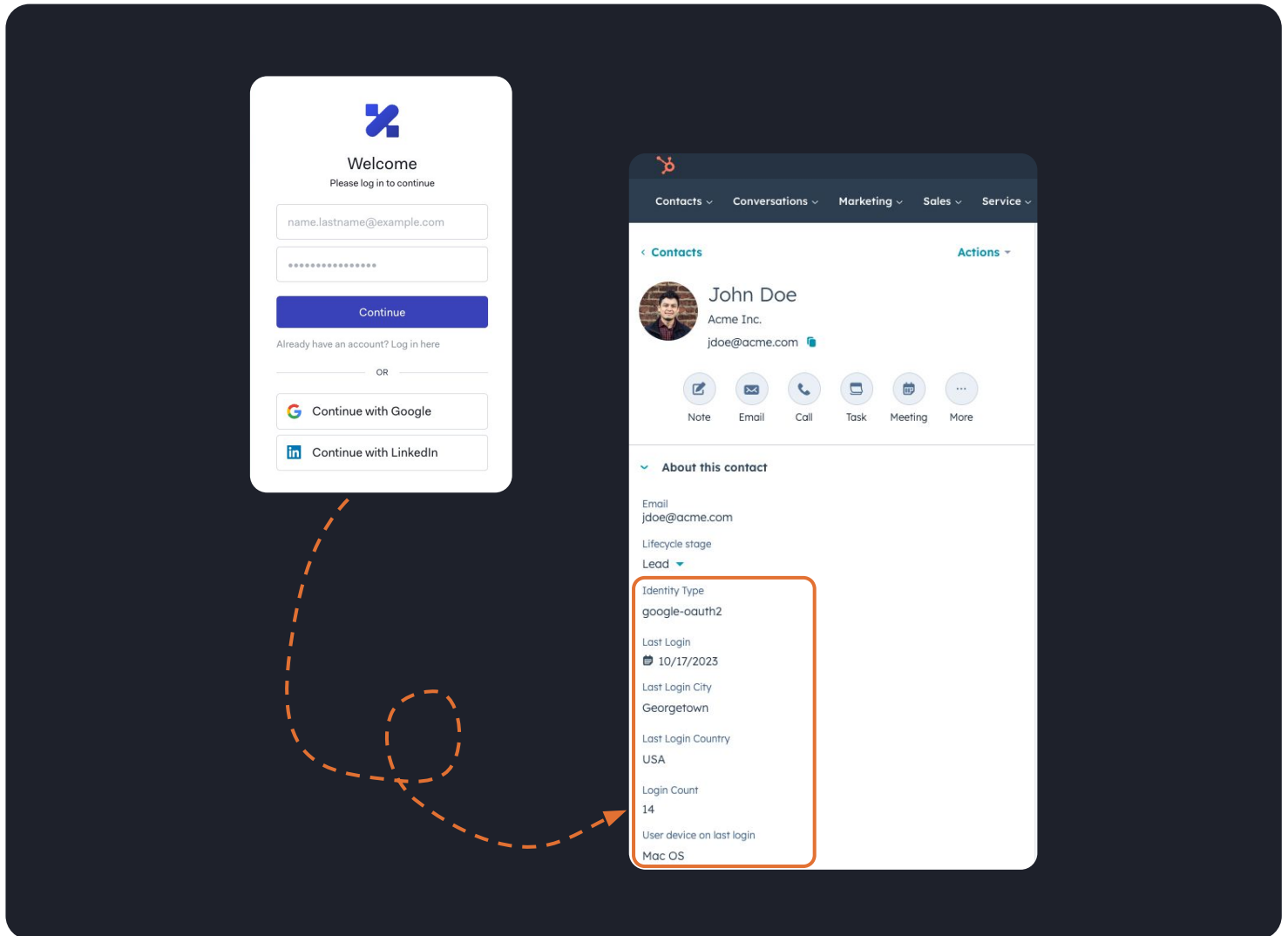
Malicious actors are just as active as organizations in finding out ways to harness the latest technology.

Data security does not have an end game, but Auth0 by Okta has powerful tools and expert personnel at your disposal to keep fraudulent activity off your platform.

What's next?

With your application secure and compliant, it's time to use CIAM to get an universal view of your consumer across multiple channels.

In our next recipe, we will explore CIAM's extensibility by [integrating your consumer data to a marketing system](#).



RECIPE 4

Send data to marketing for personalization

As organizations grow, the number of touchpoints to perform audience outreach and activation becomes exponential.

Being able to understand the journey of a single individual user across your different platforms and channels can be complicated, especially when you don't necessarily have a unified view of all your users.

SEC ★★

UX ★★★★★

Why is CIAM important for personalization?

CIAM helps understand how users are using your platform and services.

By understanding when users last logged in and from when, or even assess for inactivity, CIAM gives you data points that marketing and sales can use to promote engagement.

When your organization has CIAM, you can give your teams a better understanding of your audience, enabling personalization in many different ways — from how your product UI is presented, to marketing emails sent to your users, and whether and how they should be outreached by sales for conversion.

With focus on identity, you can ship insights of your user's journey to apps, email, and marketing systems, enabling personalized messaging just for them.

Phase 1

Ask for consent

When it comes to data privacy, CIAM helps draw a dotted line between the information your users consent to, and what is being used to personalize for them, which is vital for regulations surrounding personalization programs.

Before you can collect information about your audience and pipeline, you're going to need to ask for consent.

Check out our recipe on [data privacy compliance](#) to learn how to add consent when someone signs up for your application.

Phase 2

Unify your customer database

With Auth0 by Okta, you can connect your identity usage with your data lake or marketing automation system of choice, to have a more complete profile of each individual customer, as well as your audience as a whole.

This data can be used in many different ways — from changing the UI and suggestions in your app, to sharing information with your marketing for targeted emails, to providing analytics for product decisions.

As an example, In this recipe, we'll connect Auth0 with HubSpot, the most popular marketing automation solution for growing businesses. We will connect the dots between your users and your brand, and the integrations you use to build those customer relationships.

The integration between Auth0 by Okta and HubSpot is driven by Actions.

Actions is an orchestration platform, which gives you the freedom to add your own custom logic (and integrations) to any identity process.

You can orchestrate all sorts of processes — from syncing subscription status with Stripe, to adding identity proofing, to calling your own APIs.

Recipe

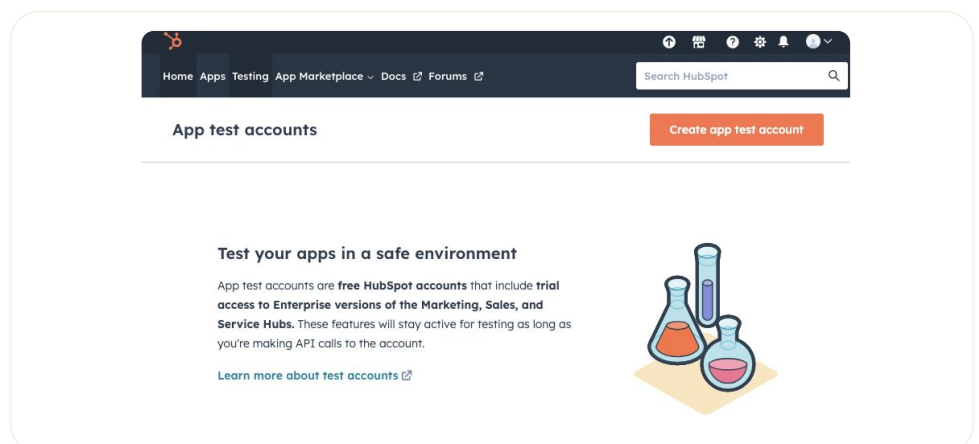
Ingredients

- HubSpot Developer Account
- Auth0 Professional or Enterprise Account
- Auth0 Actions

Configure HubSpot

Create a test environment in HubSpot

1. Access your HubSpot admin page as a Developer.
2. On HubSpot's home page, click Testing.
3. Click Create app test account.



4. Provide a name for your test environment and click Submit. The new environment will be created and displayed under the testing page.

Create an app test account

App test accounts are free HubSpot accounts that include trials of many enterprise features. [Learn more.](#)

Trial access to enterprise features will continue as long as you've made an API call to the account in the last 90 days. Any API call will reset the 90-day trial expiration window.

Account name

My app test account

CreateCancel

App test accounts

Create app test account

	ACCOUNT DETAILS	EXPIRATION DATE	CONNECTED APPS	STATUS
<input type="checkbox"/>	My app test account ID 44596242	Mar 4, 2024 3:09 PM		Active

5. Click the test environment account. You will be redirected to the HubSpot CRM dashboard for your test environment.

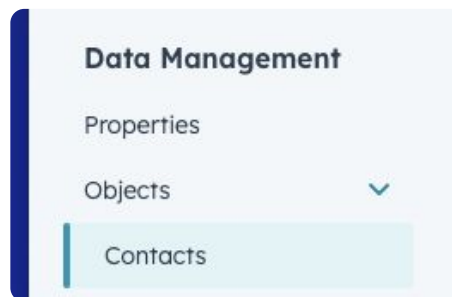
Create attributes that will receive identity data from Okta

Tip: You can create attributes to send any information available from Okta during a user login, including the user login frequency, location, device, social provider, among others. For this example, we will map the login count and city.

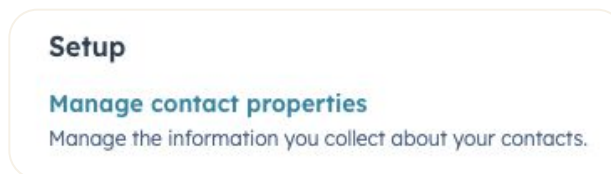
1. From the Hubspot CRM dashboard, click the gear icon on the top menu.



2. On the left bar, expand Data Management > Objects, and then click Contacts.



3. Click Manage contact properties



4. Click Manage contact properties



5. Enter the following info and click Next:
 - a. Object type: Contact
 - b. Group: Contact Activity
 - c. Label: Login Count
6. Select Number as the field type and click Next

7. On rules, make sure all options are unchecked and click Create

The screenshot shows the 'Create a new property' dialog with three tabs: BASIC INFO, FIELD TYPE, and RULES. The BASIC INFO tab is active, showing the following fields:

- Object type *: Contact
- Group *: Contact activity
- Label *: Login Count
- Description

The FIELD TYPE tab is also visible, showing:

- Field type: Number
- Number format: Formatted number
- Preview: Login Count 12,345.67

The RULES tab is active, showing the 'Select property rules' section with the following options:

- Property visibility
 - ☐ Show in forms, pop-up forms, and bots
 - ☐ Show in search results (0 of 3)
Allow users to search for information entered into this property
- Validation rules
 - ☐ Require unique values for this property (0 of 10)
 - ☐ Set min value limit
Prevent users from entering a value below a specified number
 - ☐ Set max value limit
Prevent users from entering a value above a specified number
 - ☐ Limit number of decimal places
Set the number of digits that can be entered to the right of the decimal

8. Repeat the previous steps to create the field with the following info:
- Object type: Contact
 - Group: Contact Activity
 - Label: Login City
 - Field type: Single-line text

The screenshot shows the 'Create a new property' dialog with three tabs: BASIC INFO, FIELD TYPE, and RULES. The BASIC INFO tab is active, showing the following fields:

- Object type *: Contact
- Group *: Contact activity
- Label *: Login City
- Description

The FIELD TYPE tab is also visible, showing:

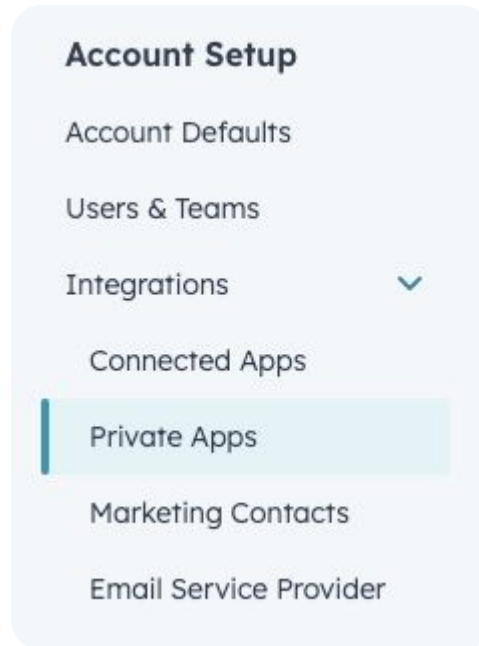
- Field type: Single-line text
- Preview: Login City Sample text

The RULES tab is active, showing the 'Select property rules' section with the following options:

- Property visibility
 - ☐ Show in forms, pop-up forms, and bots
 - ☐ Show in search results (0 of 3)
Allow users to search for information entered into this property
- Validation rules
 - ☐ Require unique values for this property (0 of 10)
 - ☐ Set min character limit
 - ☐ Set max character limit
 - ☐ Restrict to numeric values
Don't allow alpha or special characters like a, @, or \$ for this property
 - ☐ Don't allow special characters
Don't allow special characters like @, #, or & for this property

Create a HubSpot API token

9. On the left bar, expand Account Setup > Integrations, and then click Private Apps



10. On the left bar, expand Account Setup > Integrations, and then click Private Apps



- Auth0

Basic info

Scopes

Webhooks

Scopes

Scopes determine what your app can access and do in HubSpot. It's strongly encouraged to require as few scopes as possible for your app's functionality.

Selected scopes

crm.schemas.contacts.read

crm.schemas.contacts.write

crm.schemas.companies.read

Find a scope

> CMS

> CRM

> Settings

> Standard

Manage and view your CRM data

View details about property settings for contacts.
Create, delete, or make changes to property settings for contacts.

View details about property settings for companies.

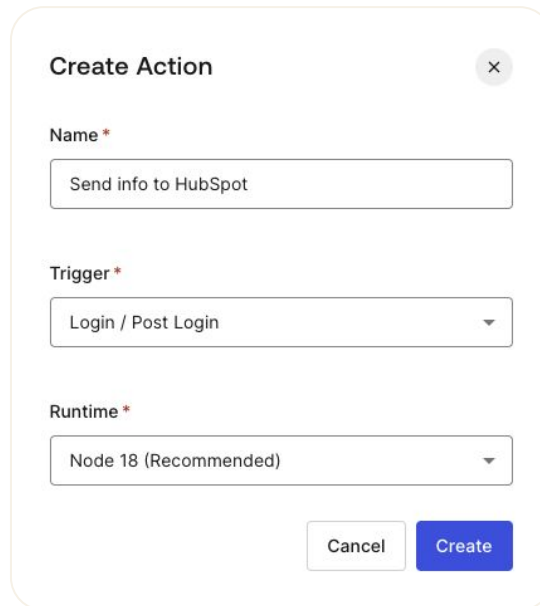
- [illegible]

Configure Auth0 by Okta Actions

Actions allow you to modify identity processes — such as registration and login — with your custom logic.

Let's create a Login Action to send data to HubSpot.

1. Navigate to **Actions > Library > Build Custom:**



The image shows a 'Create Action' dialog box with a close button (X) in the top right corner. It contains three required fields, each marked with a red asterisk:

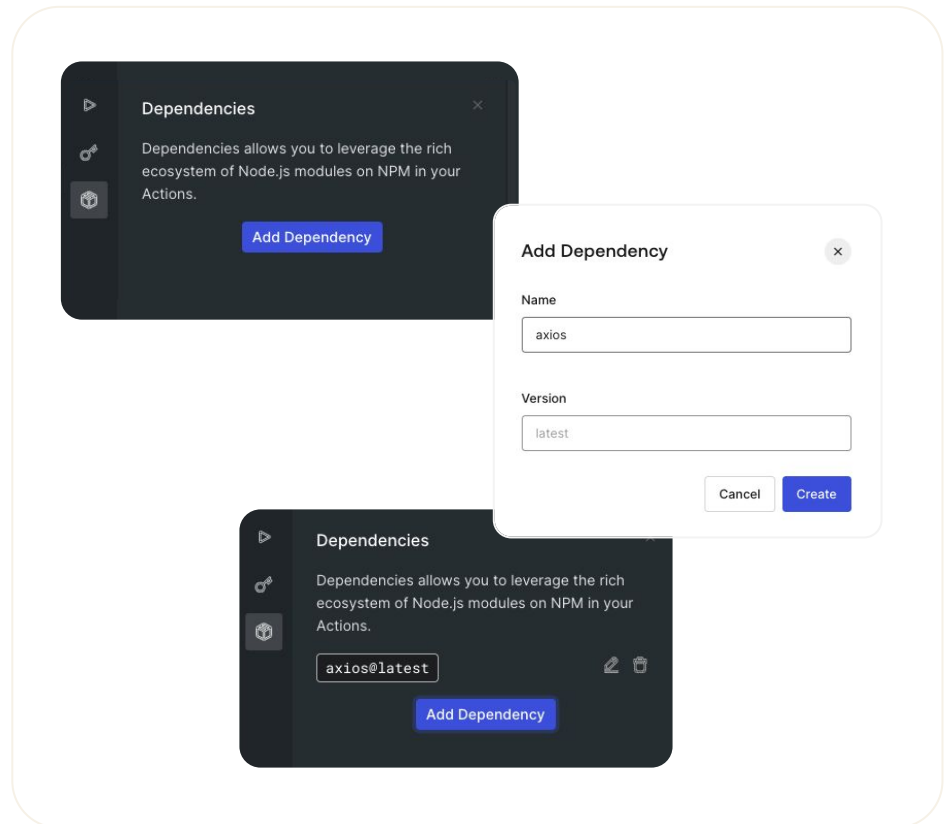
- Name ***: A text input field containing 'Send info to HubSpot'.
- Trigger ***: A dropdown menu with 'Login / Post Login' selected.
- Runtime ***: A dropdown menu with 'Node 18 (Recommended)' selected.

At the bottom right, there are two buttons: a 'Cancel' button and a blue 'Create' button.

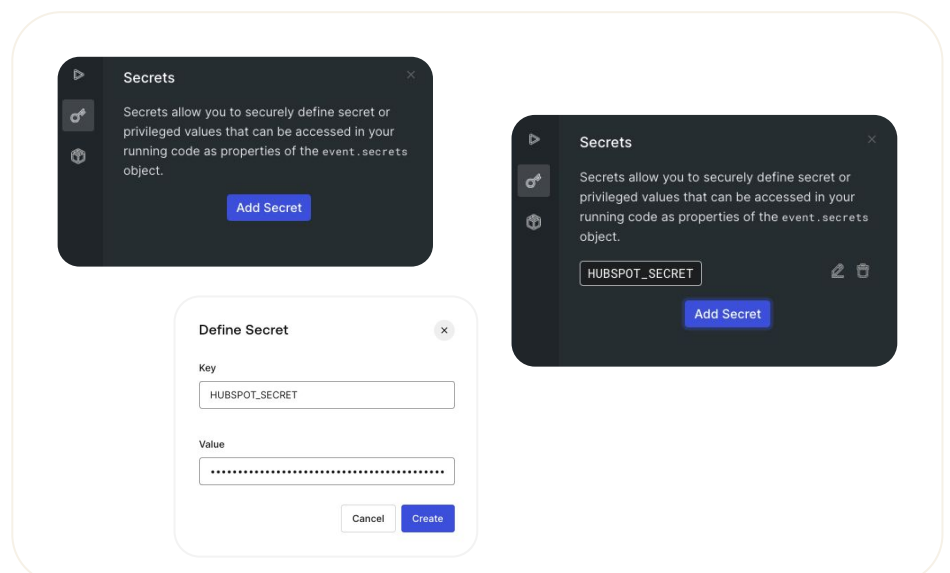
2. Here is the logic required for this example; we will be using Axios to make an HTTP call to share our user's first and last name, as well as their login frequency, and location, based on their email.

```
const axios = require("axios");  
/**  
 * Handler that will be called during the execution of a PostLogin  
 * flow.  
 */  
exports.onExecutePostLogin = async (event, api) => {  
  const url =  
    `https://api.hubapi.com/contacts/v1/contact/createOrUpdate/e  
    mail/${event.user.email}/`;  
  const headers = {  
    'Content-Type': 'application/json',  
    'Authorization': `Bearer ${event.secrets.HUBSPOT_SECRET}`  
  };  
  
  const data = {  
    properties: [  
      { property: 'firstname', value: event.user.given_name },  
      { property: 'lastname', value: event.user.family_name },  
      { property: 'login_count', value: event.stats.logins_count },  
      { property: 'login_city', value: event.request.geoip.cityName },  
    ]  
  }  
  
  await axios.post(url, JSON.stringify(data), { headers });  
};
```

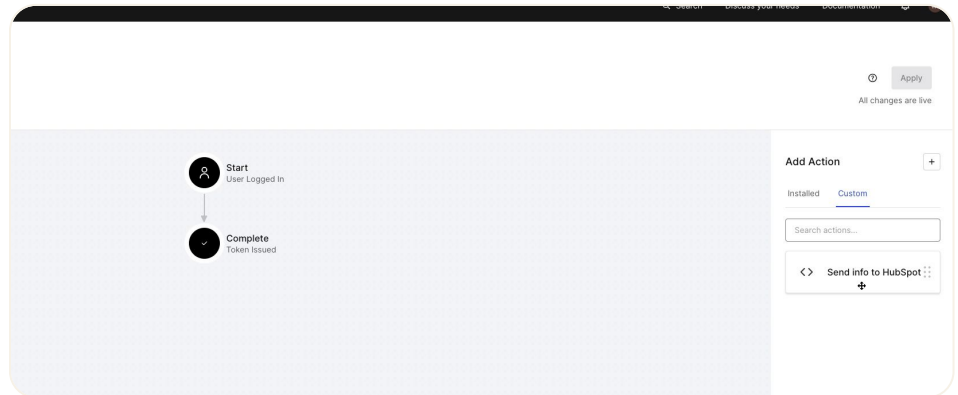

3. Add Axios as a dependency:



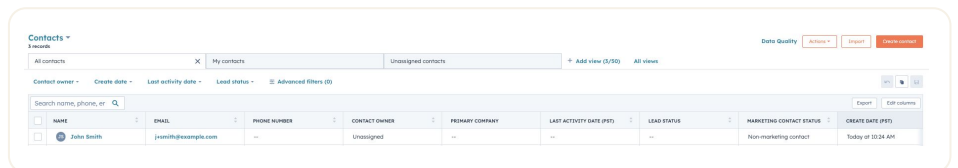
4. Add your HUBSPOT_SECRET, which is your generated HubSpot API token:



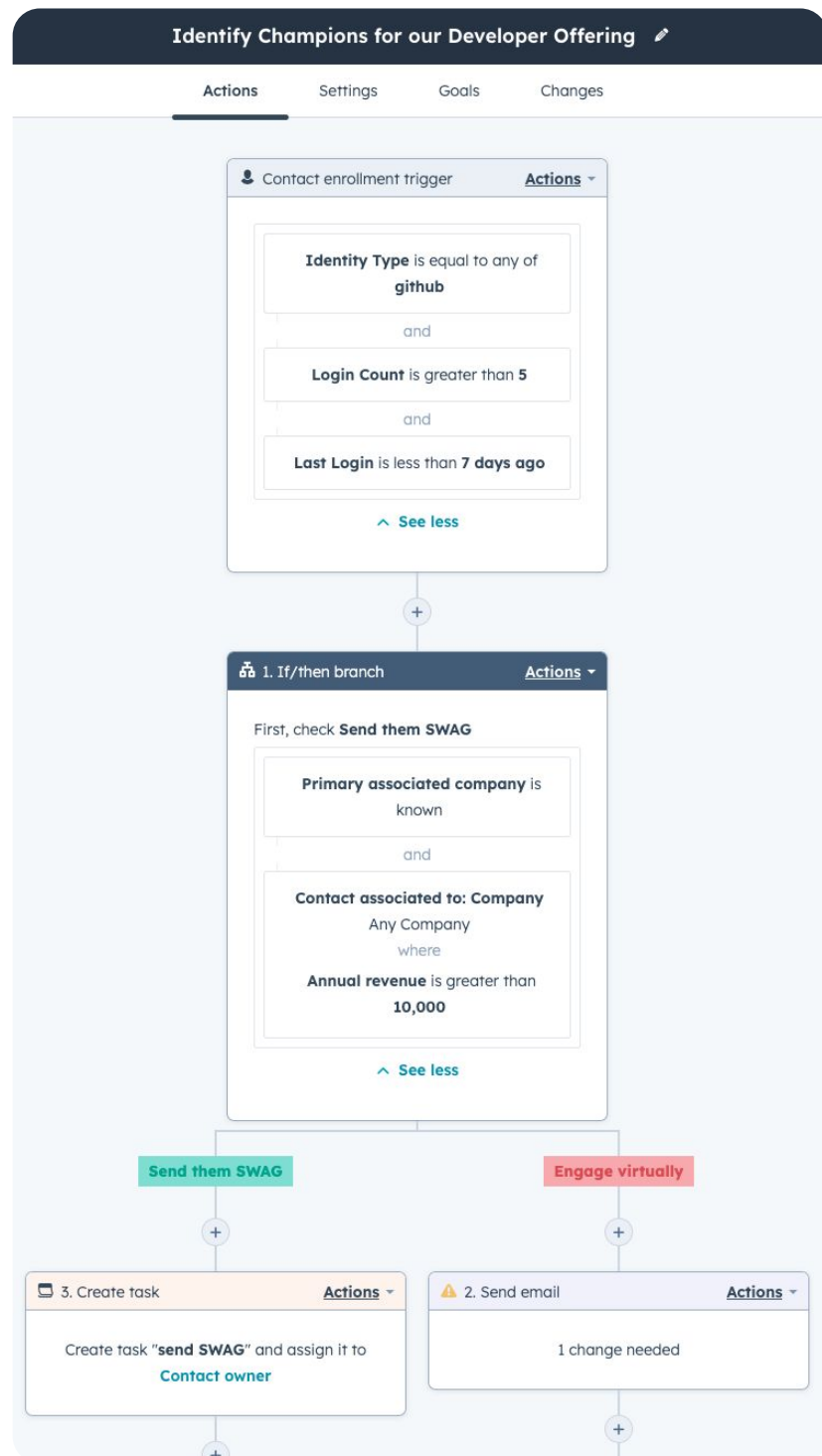
5. **Deploy** your Action to save it, then navigate to **Flows > Login**, and add your **Custom Action**:



5. Upon Successful Login or Signup (or testing your Action in the sandbox), navigate to HubSpot, and go to **Contacts > Contact** to see the fruits of your action:



Now, HubSpot can use identity data — login counts, geo, email, name, etc — in its marketing workflows to customize sales and marketing tactics. For example, send developers SWAG if they signed up with GitHub, log in with high frequency, and their accounts can be associated with a company, with good revenue:



Phase 3

Ensure your customer contacts are legit

As the influx of fraudulent activity on the web (for now) has no end in sight, fake accounts can often derail business operations and skew reporting that contribute to vital assessments for growth.

In order to ensure that your customer contacts are who they say they are, Auth0 by Okta has a number of tools to stop illegitimate traffic at the front door of your applications. Our [prevent fraudulent activities](#) recipe show how to maintain UX for your legitimate customers, while keeping bad actors — and bots — out.