

# Clarity Integration Guide

## Add a Custom Identity Verification Vendor in Okta

**Document owner:** Clarity

**Audience:** Security, IAM, and IT admins

**Applies to:** Okta Admin Console

**Purpose:** Configure Okta to use Clarity as a custom identity verification vendor for Identity Verification integration

### Important limitations

- You cannot use an IDV vendor identity provider for routing rules.
- For additional context, refer to [Okta's guidance on identity verification vendors](#) as identity providers.

---

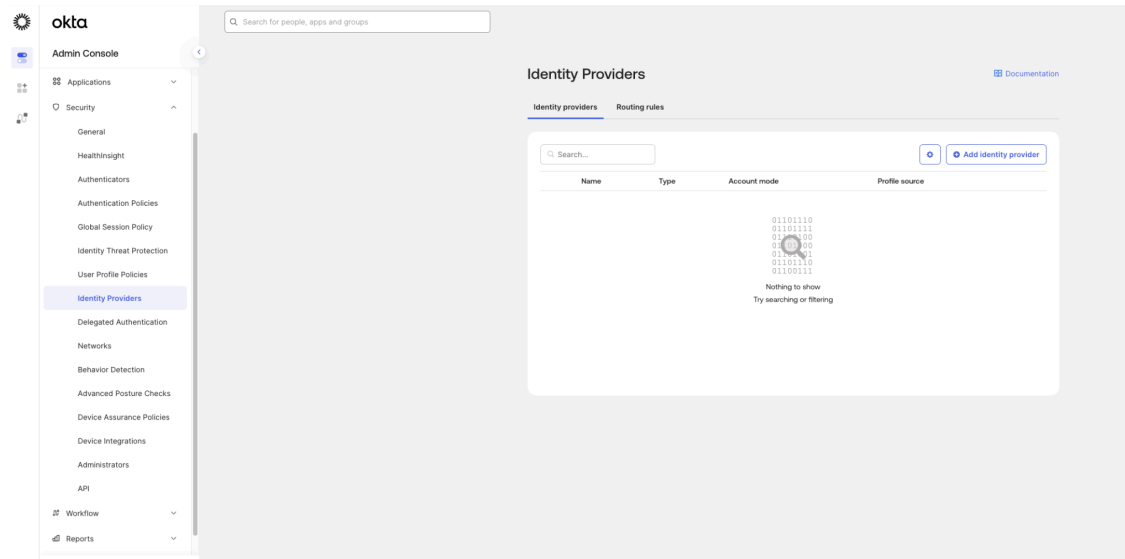
### Before you begin

1. Ensure you have Okta admin access with permissions to manage Identity Providers.

# Procedure

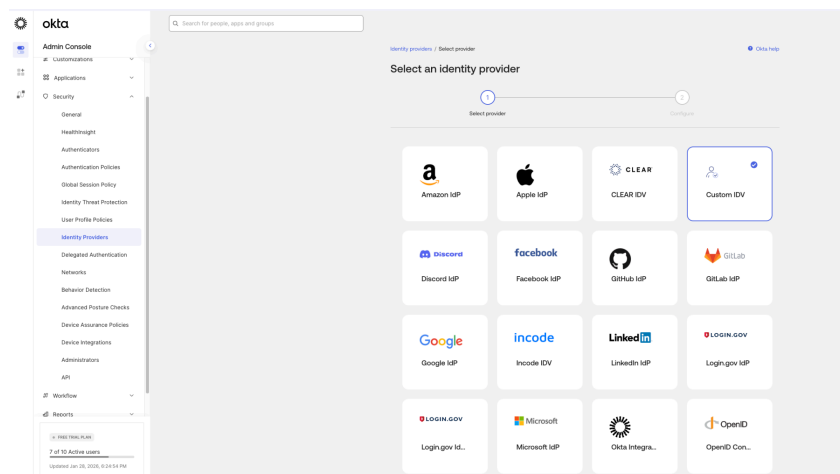
## Step 1: Navigate to Identity Providers

1. In the **Okta Admin Console**, go to **Security** → **Identity Providers**.



## Step 2: Add the Custom ID verification provider

1. Click **Add identity provider**.
2. Select **Custom ID verification**.
3. Click **Next**.



### Step 3: Configure instance details

1. In **Instance name**, enter a unique name for this integration (for example: “Clarity IDV”)

### Step 4: Configure end user sign in experience

In the **End user sign in experience** section, configure the following:

- **Vendor name**  
Enter “Clarity”. This name appears on the Sign In Widget.
- **End user license agreement URL**  
[Clarity EULA](#)  
Enter the URL for the vendor license agreement. This link appears on the Sign In Widget.
- **Privacy statement URL**  
[Clarity Privacy](#)

### Step 5: Configure Clarity’s credentials and permissions

In the **Vendor credentials and permissions** section, configure the following:

- **Client ID**  
Enter the client ID provided by Clarity.
- **Client secret**  
Enter the client secret provided by Clarity.
- **Scope**  
The following scopes are required and should be present by default:
  1. **openid**  
Required for any OIDC request and for receiving an ID token.
  2. **profile**  
Enables basic user profile info to be used in the flow, often to prefill or correlate the verification session. Okta’s own IDV guidance explicitly calls this out for IDV

integrations.

### 3. `identity_assurance`

The key scope for IDV outcomes. It requests access to identity assurance results, including the `verified_claims` structure used to convey what was verified and at what assurance level.

- Optional additional scopes:

#### 1. `email`

Useful when it is needed that the IDV flow to return an email claim and or help bind verification to a specific candidate communication channel.

#### 2. `phone`

Useful when the phone is part of the candidate identity binding, step up, or verification evidence.

The screenshot displays the Okta Admin Console interface. On the left, the 'Admin Console' sidebar is visible with a search bar and a list of navigation items: Customizations, Applications, Security, General, HealthInsight, Authenticators, Authentication Policies, Global Session Policy, Identity Threat Protection, User Profile Policies, Identity Providers (highlighted), Delegated Authentication, Networks, Behavior Detection, Advanced Posture Checks, Device Assurance Policies, Device Integrations, Administrators, API, Workflow, and Reports. The main content area is titled 'Identity Providers' and contains several sections: 'Privacy statement URL' with a text input field and a note to provide the vendor's privacy policy; 'Identity verification settings' with a sub-section 'Vendor credentials and permissions' containing fields for 'Client ID', 'Client Secret', and 'Scopes' (with 'openid' and 'profile' selected and 'identity\_assurance' added); 'Endpoints' with fields for 'Issuer', 'PAR request URL', 'Authorize URL', 'Token URL', and 'JWKS URL'; and 'User attributes' with a note about attribute mapping and a checkbox to 'Complete attribute mapping after creating the IDP instance to increase accuracy of the identity verification check'. At the bottom, there are 'Previous', 'Cancel', and 'Finish' buttons.

## Step 6: Configure endpoints

In the **Endpoints** section, configure the following:

- **Issuer**  
Enter the issuer endpoint that will be provided by Clarity.
- **PAR request URL**  
Enter the URL where Clarity handles the pushed authorization request.
- **Authorize URL**  
Enter the URL where Clarity handles the authorize request.
- **Token URL**  
Enter the URL where Clarity handles the token request.
- **JWKS URL**  
Enter the URL where Clarity provides the JSON web key set parameters used to validate the signed ID token.

## Step 7: Finish

1. Click **Finish** to save the configuration.
- 

## Validation checklist

After configuration, confirm the following:

- The Sign In Widget displays the Clarity name and links.
- The issuer, authorization, token, PAR, and JWKS endpoints match the ones provided by Clarity.
- Required scopes are present and any optional scopes are intentional.
- Your team understands that this IDV integration cannot be used for routing rules.