Tutorial

Data Residency and AI Data Protection for Okta tutorial

Table of Contents

Overview

This guide describes how to configure InCountry Data Residency integration with Okta.

The integration enables routing of Okta API traffic through InCountry infrastructure to ensure data residency compliance.

InCountry acts as a proxy layer in front of Okta and processes requests according to configured residency policies.

Okta remains the identity provider and continues to handle authentication and user management. InCountry does not act as an Identity Provider (IdP) or Service Provider (SP).

All authentication and provisioning flows are preserved and remain compatible when routed through InCountry. Requests and responses may be processed to enforce data residency requirements, including cloaking of sensitive data. Original data is securely stored within InCountry infrastructure, while cloaked data is used in communication with Okta.

Prerequisites

Before you begin, ensure the following requirements are met:

- Administrator access to Okta
- A configured Brand with a custom domain in Okta
- A configured email domain associated with the Okta Brand

You must configure two domains that share the same root domain:

1. Okta custom domain (used by Okta Brand)
2. Data Residency domain (used by InCountry proxy)

Both domains must belong to the same root domain.

Example:

- Okta custom domain: okta.example.com
- Data Residency domain: residency.example.com

The Data Residency domain must be configured with a valid SSL certificate and point to InCountry infrastructure. The InCountry service deployed on this domain acts as a proxy in front of the Okta custom domain. After completing the configuration, all Okta API requests must be sent to the Data Residency domain instead of the Okta custom domain.

Example: https://residency.example.com/oauth2/v1/token
instead of: https://okta.example.com/oauth2/v1/token

Okta remains the identity provider. InCountry acts as a proxy layer and does not modify authentication flows.

Supported Features

The integration supports standard Okta OIDC authentication and user lifecycle flows via proxy, including:

- SP-initiated SSO (Single Sign-On)
- IdP-initiated SSO (through Third-party Initiated Login)
- Just-In-Time provisioning
- SP-initiated SLO (Single Logout)

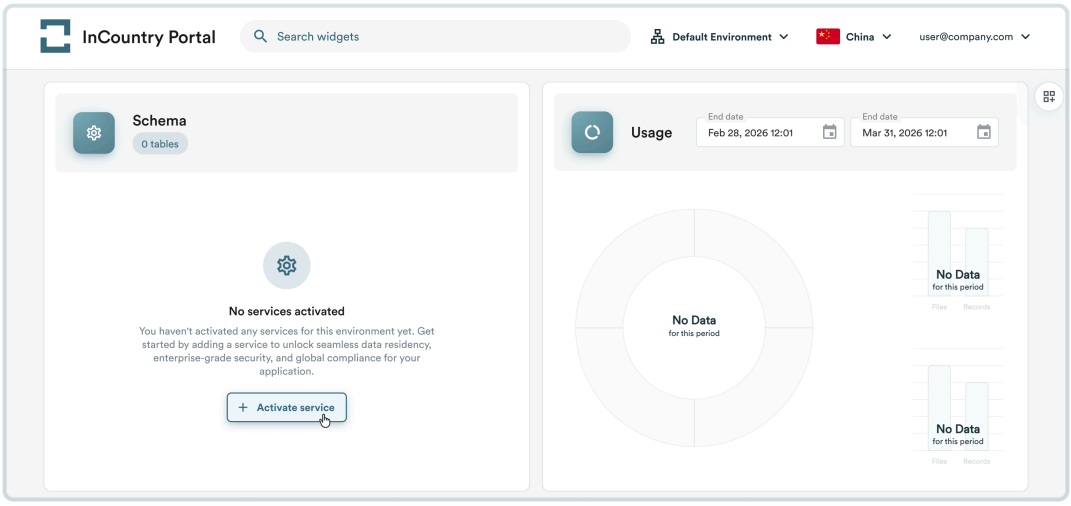
Supported Scopes

- okta.users.read
- okta.users.manage
- okta.userTypes.read
- okta.userTypes.manage
- okta.groups.read

Configuration steps

Create an Okta Service

1. Open https://portal.incountry.com/ (or https://portal.dataresidency.cn/ if you would like Data Residency in China) and create your account.
2. Select the environment or create a new one where you want to create an Okta service.
3. Select and activate the **country** for this environment where data will be managed.
4. After setting up the environment and selecting the country, you can create an Okta service. Click **Activate Service**.



5. In the opened popup, select **Okta**. Next, you will see the integration settings that need to be configured:

Domain/Server	Description
Root domain	Enter your root domain. This must be a domain that you own.
Okta Custom Domain	Enter the subdomain associated with your Okta organization. If you're unsure about Okta custom domain, refer to the Okta documentation for guidance.
Data Residency Domain	Provide your Data Residency Domain for the specific country. This domain will be used for the InCountry Data Residency service.
Authorization Server Name (optional)	Enter the Okta Authorization Server identifier. If you do not use a custom Authorization Server, leave this field empty.

Okta User Profile Attributes

1. Configure the Okta attributes that contain personal data and specify the algorithm for each.

> System attributes cannot be modified — you cannot change their algorithm or indexed parameters. Only custom attributes can be configured.

2. Enable **Indexed** for fields that you want to be searchable.
3. Click **Create**.
4. Enter the verification code sent to your email.
5. Click **Confirm**.

✅ The Okta service has been created successfully.

Configure Email Provider in the Okta Console

After the Okta service is created you will receive Okta API Base URL, OpenID Connect Discovery URL.

Server endpoint	Description
Okta API Base URL	Must be used for Okta API requests if you want user data to be cloaked and uncloaked.
OpenID Connect Discovery URL	Must be used instead of the default Okta discovery URL.

You will also receive the email provider credentials for your Okta organization.

> The configuration parameters are displayed only once. Make sure to copy them, as they cannot be retrieved later.

1. Open the **Okta Admin Console.**
2. Enable a **Custom Email Provider.**
3. After creating the Okta service, copy and paste the configuration values (**Host, Port, From address, Username, Password**) into the corresponding fields in the Okta Admin Console.

Upload Certificates for your Data Residency Domain

1. As the next step in configuring Okta, click the **Upload Certificates for your Data Residency Domain** link to upload certificates for the **Data Residency Domain** you specified earlier.

> Before uploading the TLS certificate and private key, complete the required DNS configuration and preparation steps outlined in the setup instructions on the certificate setup page.Select the TLS certificate and the certificate key.

2. Click **Upload**.

✅ The Okta service is now properly configured.

Create Okta API Service Integration

1. Open the **Okta Admin Console**
2. Go to **Applications → API Service Integrations**
3. Click **Add Integration** button
4. Find and select **Data Residency and AI Data Protection for Okta**, click Next
5. Click **Install & Authorize** button

You will receive client ID and client secret. The secret appears only once for enhanced security. Copy this secret and store it somewhere safe for use later.

How to use

You can now use the Okta Management API for users via your Data Residency domain instead of your Okta domain.

For example, to create a user with data residency, first obtain an access token using the client ID and client secret received during the Okta API Service Integration setup:

```
curl --location 'https://residency.example.com/oauth2/v1/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--header 'Authorization: Basic <base64(client_id:client_secret)>' \
--data-urlencode 'grant_type=client_credentials' \
--data-urlencode 'scope=okta.users.manage'
```

Then use the obtained access token to create a user:

```
curl --location 'https://residency.example.com/api/v1/users?activate=true' \
--header 'Accept: application/json' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer <access_token>' \
--data-raw '{
  "profile": {
    "firstName": "Ahmed",
    "lastName": "Al-Saud",
    "login": "al-saud@gmail.com",
    "email": "al-saud@gmail.com"
  }
}'
```

All profile data is securely stored in the country associated with your Data Residency domain. In Okta, this profile data appears in a cloaked form.

You should start using the InCountry Data Residency Domain instead of the Okta domain. All API calls should go through it, and all forms (hosted by Okta or not) should use it for submitting data as well. For example, use the Sign-In Page code editor in the Okta Admin Console to modify the form submit URL on the Sign-In page.